

Addressing Scheduling Software for Analyzing Delay Claim

Khalid S. Al-Gahtani

Abstract—Currently available Scheduling Software is not an optimal solution for analyzing delay claims. Such software contains features that are primarily designed for project management; which may lead to inaccuracies while processing delay claims. As a result, it is difficult to apply Delay Analysis Techniques using current scheduling software. This paper will address inaccuracies that affect current scheduling software and suggest improvements in this area of application.

Index Terms—Delay, software, scheduling, claim, construction.

I. INTRODUCTION

Most of the scheduling software packages are designed to manage project time and cost, not to measure, fix delay and acceleration responsibilities. Review of some previous studies made it possible to highlight specific uses of scheduling software that could yield unexpected results [1- 7]. These uses and issues are described in this paper. This study is useful to those who analyze delay claims; also it will serve as a resource to scheduling software developers who wish to keep their software compatible with delay analysis.

II. RELIABILITY OF THE SCHEDULE

Before using any schedule to analyze delay claim, it needs to be reliable. A summary list of items that need to be addressed before analyzing delay claims are following [2].

- All activities must be accurately and logically sequenced.
- The schedule must be complete. For example, procurement information is part of the schedule and should be included.
- The reliable schedule can be used to analyze delay even though it was not used by the contractor to complete the schedule.
- The schedule must not include any mistakes that would affect the mathematical calculations. However, any inconsequential errors in the schedule that do not affect logic or durations may be excused. Examples of such mistakes are; (1) allowing open ends activities, (2) not including in the schedule information such as a shop drawing procedure of submission, approval, fabrication, and delivery of material tied to construction activities, and (3) not

including in the schedule other trade work such as subcontractor schedule.

- The schedule must be free from the manipulation such as; (1) increasing or decreasing durations, (2) manipulating the schedule restraints, (3) manipulating the status of a job at the time a delay or change occurred, and (4) scheduling software has itself a logarithm that may result in manipulation the schedule.

More information on these errors will be explained in the following section.

III. ACTIVITY CONSTRAINTS

One purpose of scheduling is to determine the possible start and finish dates of project activities. Sometimes, the exact start and finish date can be determined by defining the predecessors and successors of each activity. In many cases the predecessor relationship is not only limitation on an activity start date; there are additional constraints that must be fulfilled before the beginning of activity. Consequently, scheduling software provide options for handling constraints on the start and finish of an activity. Constraints may include material delivery, owner permission to start work, the completion of plans, and so forth.

Improper use of these constraints can lead to unexpected results in delay analysis [2, 8]. The analyzer should examine constraints to ensure that they are realistic. Contractual constraints, which are linked to requirements in the contract, should be distinguished from other constraints [8]. The reviewer should know whether constraints are based on the contract agreement or used only to influence the analysis result. Using the constraints to consume float and force some activity to be critical is a well known practice [2]. As a result, these constraints, if not considered properly, can be potentially dangerous for delay analysis.

IV. DEFINING MULTIPLE CALENDARS

Current scheduling software packages allow different calendars for each activity to make the schedule realistic. In some situations, the use of different calendars can yield inaccurate results for delay analysis [1, 3, 6]. Additionally, it can affect the outcome of the Total Float analysis. These inaccurate results occur because the software does not define the responsibilities of the non-working days on the calendars.

Most of the time, these periods are contractually determined and considered as Excusable Non compensable delay, for which neither the owner nor the contractor should be responsible. Therefore, unlinking the responsibilities from

Manuscript received May 17, 2012; revised June 18, 2012.

Khalid S. Al-Gahtani is with College of Engineering, Civil Engineering Department, King Saud University, Riyadh, 11421, Saudi Arabia(kgahtani@ksu.edu.sa).

the calendar allows manipulation in the scheduling software.

V. LATE DATES SCHEDULING

Some software includes a feature that schedules the start and finish of all activities based on the late dates, not the early dates [2]. When this occurs, all the activities are forced to be critical. These options should be avoided because it leads to completely inaccurate delay analysis. A schedule that is manipulated so that noncritical activities are considered critical while identifying delay events is called “Spring-Loaded” [2]. Four check points are suggested to identify such manipulations:

- Are activity durations all multiples of 5-day or 7-day weeks? This indicates that there is no effort to identify the activity durations accurately.
- Is the percentage of critical to noncritical activities more than 10%?
- Are constraints manipulated to force noncritical activities to be critical?
- Is the average float per activity higher than usual [2]?

VI. IMPROPER SCHEDULE UPDATE

Most of the current software updates the schedule to track the actual progress to manage the project accurately. According to the available as-built data, scheduling software can update the schedule by many ways. When any of the as-built data is missing, the schedule can be updated with unexpected results. Three areas exist in which schedule update errors can lead to erroneous measurements of delay [1, 2, and 9]. These areas are briefly discussed below.

A. Actual Date Manipulation

When updating the schedule, all the early and late dates must be converted to actual dates to document the as-built data. However, making incorrect actual start or finish dates can cause an unexpected delay analysis result. In addition, changing the actual dates is a recognized manner in which contractors can manipulate the schedule in the delay claims [9]. To avoid such errors, all the actual data should be removed in analyzing the delay claim [2]. Further on this solution, the delay types should be incorporated in the software beside the actual data. To best of our knowledge none of the current software identifies the delay and acceleration types (Excusable/Compensable delay, and Owner/Contractor’s Acceleration) of each activity’s duration. Therefore a scheduler that ignores the delay and acceleration types in the activity durations, as well as ignoring the calendars non-working periods, is unable to analyze the delay claims accurately.

B. “Out of Sequence” Activities

When a schedule is updated with information for intermediate progressed activities that have already started, and the activity is still in progress, the project completion dates, which delay claim analysis relies on, might be affected. In general, all the activity dates that fall after the schedule-update should be estimated as early and finish dates. Moreover, all the activity dates that fall before the

schedule-update should be actual dates. Intermediate progressed activities have actual start and early finish because they are still in progress.

Depending on the scheduling software used, there are two ways to determine the Early Finish for the intermediate activities [2, 10]. Early Finish date is required in order to update the schedule, which can result to “out of sequence” situation as explained below.

The term “Out of Sequence” refers two activities that are planned to start in a sequence but which actually start simultaneously with overlap in order to accelerate the project. As a result, the successor activity starts actually “out of sequence” [2, 11]. The current Scheduling software updates the schedule in such situations by following one of two methodologies: “Retained logic” or “Progress override” [1]. These two methods either retain the original sequence logic (retained logic) or allow progress to override the old sequence logic (progress override).

Selection of Retained Logic allows the out of sequence activity to start early, but schedule its completion according to the sequence logic. The activity will not be allowed to complete until all its predecessors are finished, but Progress Override treats the out of sequence activity as it has no predecessors so it can continue without affected by its unfinished predecessors.

Fig. 1 shows completion of activity B (out of sequence) differs in (a) and (b) depending on methodology used for update schedule. It was also shown by Arditi and Pattanakitchamroon that these two methodologies can lead to different project completion dates [1], consequently, the delay analysis might be affected. Both options are needed to reflect the actual situation on the project. However, using the “progress override” option with a relation that depends only on logic constraint (not a resource constraint) can lead to unexpected result.

In the sake of analyzing delay claim, it needs to pay attention to the out of sequence activities in order to not cause any error in determining the project time. Furthermore, the responsibility of accelerating the schedule because of using out of sequence options in the scheduling software should be defined and included in the delay analysis.

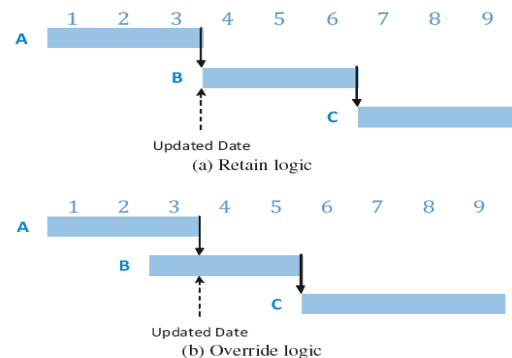


Fig. 1. Out of sequence situation

VII. DANGLING ACTIVITIES

A “dangling” activity is an activity that is not tied to any successor activities. Scheduling software marks the status of these activities as critical, because they are independent from the rest of the activities on the schedule [2]. However, this

assumption is not realistic, because the delay on a dangling activity may not necessarily delay the project completion. Determining the total float of dangling activities is difficult; it is therefore recommended that they are to be tied with the project start and finish, to enable the calculation of the total float of dangling activities [2].

It is possible for an activity that is tied to a start and finish date to become a "dangling" activity, sometimes without the knowledge of the scheduler. Hobb described scheduling practices that can lead to such dangling activities [5]. These dangling situations occur when Start-to-Start and Finish-to-Finish relationships are used. Hobb explains that such activities have negative consequences for the delay analysis, because they are isolated from their float path and their criticalities are difficult to determine. In order to avoid these problems, Hobb recommended using the Finish-to-Start relationship [5].

VIII. NEGATIVE FLOAT

In project scheduling, float is always zero or a positive number. Negative float values have been introduced recently to indicate that not only is an activity critical but that it delays the project by a negative number of days on the total float [1, 9, and 12]. The effect of negative float in delay analysis is that some scheduling software uses the negative total float as an indication of whether the activity is critical. Arditi and Pattanakitcharoom prove with an example that, in some situations, the negative float can be a poor indicator of critical activities [1]. There are two approaches to determining the critical path. The first approach is based on the total float value (if it is zero or negative); and the second approach is based on the longest path. In a situations of concurrent delay or a path change from critical to noncritical, the two approaches give different results [1].

IX. DIFFERENT USERS

The last source of error that contributes to unexpected analysis results is the use of an incorrect practice to create as-planned schedule. In general, the schedule should reflect the actual plan in order to be used as legal document in the courts [2]. Not only should the schedule be realistic in the as-planned stage, but it should also be able to be updated without any error. For example, using illogical constraints might not always affect the as-planned schedule, however, when the schedule is updated, this constraint could yield an incorrect result. Studies that attempt to develop accurate delay analysis results emphasize the importance of making a good baseline schedule [4, 7]. Winter developed 110 checks to use when reviewing the baseline schedule [7]. Popescu-Kohler described a flowchart process for the baseline schedule, with the involvement of the project team [13]. Additionally, Arditi and Pattanakitchamroon identified three incorrect practices that should be avoided [1]:

- Unreasonable use of constraints,
- Unconventional logic such as Start-to-Finish, and
- Long or negative lag times [4].

In addition, Becica et al. point out that the Finish-to-Finish

(FF) relationship should be avoided in delay analysis, because there is no legal root for FF relation [8]. They suggested that the positive FF lag should be removed when conducting analysis using the As-planned impact method, but not when conducting "but for" analysis [8]. Al-Gahtani introduces a list of equations that can be used to check the delay on the schedule [14]. Delay of an activity may occur in one of two forms: delay in activity duration or delay in the start date. These equations are tested by using current popular software that can produce a working day time for the actual start and finish for each activity. The following are the proposed equations:

A. *Delay on an Activity duration (with ZERO day time convention):*

$$\text{Activity Delay} = \text{AF} - \text{AS} - \text{OD} \quad (1)$$

where

AF = Actual Finish for the activity. The unit should be working day (not a calendar date) in order to avoid the weekend and non-working days.

AS = Actual Start for the activity (working day).

OD = Original Duration for activity (working day).

B. *For Relation logic (With ZERO day time convention):*

▪ *Finish to Start (FS):*

$$\text{Theoretical Activity Start} = \text{AF}_p + \text{Lag} \quad (2)$$

where

AF_p = Actual Finish of predecessor activity (working day).

Lag = relation lag duration (working day).

▪ *Finish to Finish (FF):*

$$\text{Theoretical Activity Start} = \text{AF}_p + \text{Lag} - [\text{AF}_s - \text{AS}_s] \quad (3)$$

where

AF_s = Actual Start of successor activity (working day).

AS_s = Actual Start of successor activity (working day).

▪ *Start to Finish (SF):*

$$\text{Theoretical Activity Start} = \text{AS}_p + \text{lag} - [\text{AF}_p - \text{AS}_s] \quad (4)$$

where

AS_p = Actual Start of predecessor activity (working day).

▪ *Start to Start (SS):*

$$\text{Theoretical Activity Start} = \text{AS}_p + \text{lag} \quad (5)$$

where

AS_p = Actual Start of predecessor activity (working day).

$$\text{Lag Delay} = \text{AS}_s - \text{Maximum values of Theoretical Activity Starts} \quad (6)$$

X. CONCLUSION

This paper introduces some concerns about the use of scheduling software. The first part of the paper identifies seven areas of concern, which the user should be aware of. Some solutions are suggested to avoid inaccuracy while using scheduling software in its analysis. Therefore, the ideal scheduling software must:

- Track the schedule progress time and cost
- Fix the project delay/acceleration responsibilities
- Track the total float responsibilities

- Determine the damages in time and cost for all project participants.

The major recommendation is that the scheduling software should be adjusted to include type of delays on the activity and its duration, in addition to define it on the software package calendar. Other recommendation is to avoid incorrect procedures by making a checklist of these practices and ensure that the schedule is free of them.

ACKNOWLEDGMENT

This project was supported by King Saud University, Deanship of Scientific Research, College of Engineering Research Center. The authors would like to thank King Saud University, Riyadh, Saudi Arabia, for providing various facilities for conducting this study.

REFERENCES

- [1] D. Arditi and T. Pattanakitchamroon, "Selecting a delay analysis method in resolving construction claims," *International Journal of Project Management*, vol. 24, no. 2, pp. 145-155, February, 2006.
- [2] M. T. Callahan, D. G. Quackenbush, and J. E. Rowings, *Construction project scheduling*, McGraw-Hill, New York. 1992.
- [3] K. Kim and J. M. De La Garza, "Critical path method with multiple calendars," *Journal of Construction Engineering & Management-ASCE*, vol. 131, no. 3, pp. 330-342, 2005.
- [4] R. Korman, 2003. "Critics Can't Find the Logic in Many of Today's CPM Schedules." PMI, College of Scheduling. [Online]. Available: <http://www.pmicos.org/fse.asp>.
- [5] L. J. Hobb, "There's more than carrots dangling here," *Cost Engineering*, vol. 46, no. 05, pp. 7-8, 2004.
- [6] N. J. Scavino, "Effect of multiple calendars on total float and critical path," *Cost Engineering*, vol. 45, no. 06, pp. 11-15, 2003.
- [7] R. Winter, "Checks and Balances: Baseline Schedule Review," 2002 *Primavera User Conference*, San Diego, CA.
- [8] M. Becica, E. R. Scott, and A. B. Willett, "Evaluating responsibility for schedule delays on utility construction projects," *Proceedings of the 53rd Annual Meeting of the American Power Conference*, Chicago, IL, USA, 893-896.
- [9] J. G. Zack, "Schedule 'games' people play, and some suggested 'remedies'," *Journal of Management in Engineering*, vol. 8, no. 2, pp. 138-152, 1992.
- [10] T. F. Peters, "Dissecting the Doctrine of Concurrent Delay," 2003 *AACE International Transactions*, Orlando, FL, United States, CDR011-CDR018
- [11] P. Levin, *Construction Contract Claims, Changes and Dispute Resolution*, ASCE Press, Reston, VA. 1998.
- [12] T. J. Trauner, *Construction Delays: Documenting Causes, Winning Claims, Recovering Costs*, R.S. Means Co., Kingston, MA. 1990.
- [13] A. I. Popescu-Kohler, "Improvements to construction delay analysis techniques," Ph. D. Dissertation, University of Texas at Austin, Austin, Texas. 1998.
- [14] K. S. Al-Gahtani, "Construction Delay Claim Analysis by using Expert System," Master theses, King Saud University, Riyadh, Saudi Arabia. 1999.



Khalid S. Al-Gahtani got his B.Sc. and Master degree in Civil Engineering from King Saud University (KSU). He received his PhD degree from State University of New York at Buffalo in Construction Engineering and Management. He is an Assistant professor for Civil Engineering in KSU, and Adjunct Assistant Professor in Missouri University for S&T. He is a member in many associations including Association for the Advancement of Cost Engineering International (AACE) International, Saudi Council of Engineers, American Association of Civil Engineering, (ASCE), and International Association of Computer Science and Information Technology(IACSIT). He is also an affiliate of different organizations. He is author of reference book in the field of Construction Law and he translated a reference book in this field from English to Arabic. In addition, he has number of publications in international journals.