The Study on Methodological Framework for Comparing Delay Analysis Methods in Construction Contracts

by

Alena Vasilyeva-Lyulina

A dissertation submitted in satisfaction
with the requirements for the degree of

Doctor of Philosophy
in
Engineering

in the
GRADUATE SCHOOL OF ENGINEERING
DEPARTMENT OF URBAN MANAGEMENT
of
KYOTO UNIVERSITY

2013

The Study on Methodological Framework for Comparing Delay Analysis Methods in Construction Contracts

by

Alena Vasilyeva-Lyulina

Acknowledgments

I would like to offer my sincerest gratitude to my supervisor, Professor Kiyoshi Kobayshi, who accepted me in his laboratory, guided and inspired through all these years, putting challenges and never doubting my ability to overcome them.

I would like to thank Professor Kakuya Matsushima for his help during my entrance exams, guidance and assistance with the research during laboratory seminars and as a member of defense committee.

I would like to address special thanks to Professor Masamitsu Onishi, who accompanied me along this interesting and uneasy journey, helping to stay on track and do my best.

This research would have never been conducted without Professor Toshihiko Omoto, who inspired me to choose this topic and constantly supported during the study.

I would like to thank Ms. Marianne C. Ramey for unforgettable experience and inspiration I gained during my internships in Bucharest and London.

I am very indebted to my colleagues and laboratory members, who were always there ready to help with any problems related to the studies or not, especially Aya Fujimoto, Hayeong Jeong, Ismu Rini Dwi Ari, Thu Hằng Bùi, Nakano Hitedetoshi, Maiku Abe, Kenshiro Ogi and many others.

Four years of study in Japan was an amazing time with its bright periods and moment of struggle. I would have never made it without support of my family, my brothers Alexey and Alexander, and my best friends back in Russia - Anna P. and Katherine G.

I would like to address special thanks to Raisa Ivanovna Hramtsova, who gave me the strengths to believe in myself and without whom I would have never dare to get that far from home.

I would like to thank my friends in Japan Ilona A., Lena B., Victor G., Olga G., Rafael A., Tyana S., Miguel E., Tofig H., Romanas Ch., Daniela M. and many-many others.

I also thank the Ministry of Education, Culture, Sports, Science and Technology of Japan - Japanese Monbukagakusho - for providing this amazing opportunity to live in Japan and touch its unique culture.

Finally, I would like to thank God for feeling blessed, for all miracles that happened to me during these years, for all wonderful people I have met and have an honor to call friends or to whom I can now express my deepest gratitude.

Abstracts

Nearly every construction project experiences time overrun beyond originally planned dates. Significant delays damage the project and often result in additional cost, payment withhold and demand for liquidated damages.

In order to establish responsibility for the time and cost overrun all project records are gathered and carefully investigated. However even if available information, project records and schedules are complete and consistent, knowing the facts does not mean the responsibility can be immediately allocated among the parties. Project delay is complex in nature; it is caused by various happenings and factors influencing at different periods of time and attributed to different parties.

To establish this causation and to allocate liability among the parties further thorough investigation is required. In modern project management practice it is called Delay Analysis (DA).

Numerous studies have been devoted to the DA methods comparison. However behind that lies another fundamental problem – confusion and lack of body of knowledge or standard in definition of delay concept itself and methods' algorithms. For example such well known method as Time Impact Analysis (TIA) can be understood differently in terms what delay event is, in which order they should be added, etc. In fact most of delay analyses in practice are not based on any methodology at all, but just on expert's logic and experience.

In this environment of the lack of clear definition of DA establishing of methodological framework and criteria of evaluation of delay analysis methods, its critical principles and desirability, becomes an important subject of research.

The purpose of the following research is to establish the methodological framework for delay analysis, to formalize existing conventional techniques and to improve methodology by defining the key elements of delay analysis and its application rules.

Schedule delays are a threat to any kind of project. Delay Analysis same as Project Management Methodology can be applied in various industries, however depending on a type of project the implementation of method and practices would have its own properties and peculiarities. The following research concentrates on forensic delay analysis in the context of construction industry and related disputes. Specifically on the

delay analysis employing the critical path method (CPM) scheduling, as the dominant practice in construction project planning and execution.

The dissertation contains six chapters. The first half of the thesis present the research background and literature review. The second half devoted to achievement of the research objectives and present its findings.

Namely, Chapter 2 explains basic terms and concepts of Delay Analysis and discusses the reasons behind its complexity and justifies the necessity of the research.

Chapter 3 is devoted to the literature review on existing conventional methods and their mathematical formalization using unified language, created as one of the contributions on the present study.

Chapter 4 investigates the key elements of delay analysis, which form its methodological framework and algorithms. Desirable properties of delay events and delay analysis options are established and discussed.

Chapter 5 formulates the delay analysis problem with consideration of the network structure and proves the existence of core of the proposed liability allocation game.

Chapter 6 concludes the study and proposes some potential future research topics.

Appendix A contains proves of several propositions, made in chapter 5.

List of Tables

List of Figures

Contents

Chapter 1

INTRODUCTION

1.1 Background

Nearly every construction project experiences time overrun beyond originally planned dates. Significant delays damage the project and often result in additional cost, payment withhold and demand for liquidated damages. Several studies have shown that contract schedule and payment issues are two most common items of dispute in construction [16, 28, 92, 59, 60].

For example, In the United Kingdom, a 2001 report by the National Audit Office, entitled “Modernizing Construction” revealed that 70% of the projects undertaken by government departments and agencies were delivered late [68]. Assaf and Al-Hejji (2006) did a survey on time performance of different types of construction projects in Saudi Arabia ran into the same conclusion (45 out of 76 projects were delayed) [9]. A study conducted by the Infrastructure and Project Monitoring Division of the Ministry of Statistics and Programme Implementation in 2004 in India reported that of 646 central sector projects, costing about $50 trillion (U.S. dollars), approximately 40% were behind schedule, with delays ranging up to 252 months [49].

In order to establish responsibility for the time and cost overrun all project records are gathered and carefully investigated. However even if available information, project records and schedules are complete and consistent, knowing the facts does not mean the responsibility can be immediately allocated among the parties. Project delay is complex in nature; it is caused by various happenings and factors influencing at different periods of time and attributed to different parties.

According to Tieder (2009) it is an accepted law in US, that to recover for delay a party must prove three elements: (i) liability of the party against whom it is making the claim, (ii) causation and (iii) damage. [98]. In other words it is needed to prove that particular delay event was a cause of some certain damage to the project and is a responsibility of the other party. Tieder also confirms that "proof of liability and damage alone do not entitle a party to recovery" [98]. Hence the proof of causation, proof that the event in dispute is the source of the claimed damage becomes essential.

To establish this causation and to allocate liability among the parties further thorough investigation is required. In modern project management practice it is called Delay Analysis (DA).

Ndekugri et.al (2008) defined DA as "the task of investigating the events that led to project delay for the purpose of determining the financial responsibilities of the contracting parties arising from the delay [77]. DA has developed as a means of providing the justification and quantication of the time and/or cost consequences necessary for resolving the different contentions [17]. DA refers to a forensic investigation into what has caused a project to run late [33].

However even though DA is most often performed during and for disputes, analysis itself can be done at any stage of the project for different kind of purposes, including further delay damage predictions or analysis of past experience. Moreover the discussed practices application is not limited to construction projects and can be used for any other type of the project, which experiences delay. But in the present study DA is assessed within a context of construction dispute resolution.

From the common sense point of view the analysis of "what has cause a project to run late" should be an investigation into what delaying events have actually occurred during the project. However, being a huge and important task itself, data gathering and establishing facts is only part of the problem. From methodological point of view, the focus of DA is not on getting reliable information, but rather on how to deal with it, in other words how to assess the contribution of particular happening(s) to the project delay in completion.

Over past twenty years several DA methods or techniques have been created. There is no certain number of how many of them exist today, but for example J.B. Yang et.al (2009) reviewed 18 DA methods [103].

Each method has its strong points, but application of two methods to the same project usually results in different if not contrary conclusions. And this in turn instead of clarification just leads to a new argument which DA is more reliable and which technique is preferable.

Numerous studies have been devoted to the DA methods comparison [7, 11, 16, 22, 25, 32, 51, 73, 77]. However behind that lies another fundamental problem confusion and lack of body of knowledge or standard in definition of delay concept itself and methods' algorithms. For example such well known method as Time Impact Analysis (TIA) can be understood differently in terms what delay event is, in which order they should be added, etc. In fact most of delay analyses in practice are not based on any methodology at all, but just on expert's logic and experience.

In this environment of the lack of clear definition of DA establishing of methodological framework and criteria of evaluation of delay analysis methods, its critical principles and desirability, becomes an important subject of research.

1.2 Objective and Scope of the Study

The purpose of the following research is to establish the methodological framework for delay analysis, to formalize existing conventional techniques and to improve methodology by defining the key elements of delay analysis and its application rules.

To achieve it goals the specific objectives are chosen:

1. To reveal the lack of unified approach and formulations in delay analysis theory;

2. To review existing delay analysis methods, its interpretations, variations and assessment of strong and weak points by different researchers;

3. To formalize conventional delay analysis methods through developing of unified mathematical language;

4. To extend the concept of delay event, as a key element of delay analysis, define the desirable properties of delay events and evaluate existing DA techniques in terms of correspondence to them.

5. To consider 'bottle necks' or points, where delay analysis becomes fraud with subjectivity and manipulations. Define the desirable solutions for each weak spot.

6. To develop a formal structure of delay analysis problem;

7. To investigate whether DA methods have any tendency in the analyzed results;
8. To address fairness issue of delay analysis methods;
9. To draw conclusions and put suggestions for the future research.

Schedule delays are a threat to any kind of project. Delay Analysis same as Project Management Methodology can be applied in various industries, however depending on a type of project the implementation of method and practices would have its own properties and peculiarities. The following research concentrates on forensic delay analysis in the context of construction industry and related disputes. Specifically on the delay analysis employing the critical path method (CPM) scheduling, as the dominant practice in construction project planning and execution.

1.3 Methodology of the Study

To achieved the objectives of the study, first the investigation of existing delay analysis methods and concepts was conducted. Gathered materials were analyzed and reviewed in chapter two and three, also using the method of case study. Second, to formalize the reviewed techniques methods of mathematical analysis were applied, including theory of sets and functions. Third, method of inductive and deductive reasoning were used to enrich the concept of delay event and identify the key elements of desirable delay analysis method. Forth, to formulate the delay analysis problem cooperative game theory is applied.

1.4 Structure of the Dissertation

The dissertation contains six chapters. The first half of the thesis present the research background and literature review. The second half devoted to achievement of the research objectives and present its findings.

Namely, Chapter 2 explains basic terms and concepts of Delay Analysis and discusses the reasons behind its complexity and justifies the necessity of the research.

Chapter 3 is devoted to the literature review on existing conventional methods and their mathematical formalization using unified language, created as one of the contributions on the present study.

Chapter 4 investigates the key elements of delay analysis, which form its methodological framework and algorithms. Desirable properties of delay events and delay analysis options are established and discussed.

Chapter 5 formulates the delay analysis problem with consideration of the network structure and proves the existance of core of the proposed liability allocation game.

Chapter 6 concludes the study and proposes some potential future research topics.

Appendix A contains proves of several propositions, made in chapter 5.

Chapter 2

DELAY ANALYSIS IN CONSTRUCTION INDUSTRY

2.1 Impact of Delay Event on completion date

2.1.1 Definition of Delay

In the Cambridge Advanced Learner's Dictionary (2011) *delay* is defined as " to make something happen at a later time than originally planned or expected" [23]. Merriam-Webster Online Dictionary (2012) gives four definitions of delay: (i) the act of delaying, (ii) the state of being delayed; (iii) an instance of being delayed; and (iv) the time during which something is delayed. [101]

In the project management practice, a delay is used to reflect "the time overrun either beyond completion date specified in a contract, or beyond the date that the parties agreed upon for delivery of a project" [9]. It is an act or event that extends the time necessary to finish activities under a contract [96].

However despite the intuitively understandable definition, there are some difficulties with the term. As K.Pickavance (2005) claimed that in the construction and engineering industries there is no standard form of the contract in which it is defined and the word *delay* is "from time to time used to mean 23 different things". The reason suggested by the author is that it has no an "intrinsic quality" of its own, and to define its meaning it should be first related to something else. But in the same work K.Pickavance (2005) concludes that "for a given work content **"delay" is really no more than the difference between**

an intention and reality" [82]

This difference can be attributed to a single activity, to group of activities or to the whole Project. *Delay to progress* would be used to mark activity delay and *Delay to completion* - to note delay to the Time for Completion [1].

Delay should not be confused with a term "delay event", which is discussed in the section 2.1.4 "Delay Event".

2.1.2 Project Programme and Schedules

Clause 8.3 of the FIDIC Red Book (1999) states that: "The Contractor shall submit a detailed time programme to the Engineer within 28 days after receiving the notice under Sub-Clause 8.1 [Commencement of Works]. The contractor shall also submit a revised programme whenever the previous programme is inconsistent with actual progress or with the Contractor's obligation. Each programme shall include... the order in which the Contractor intends to carry out the Works, including the anticipating timing of each stage..." [34].

In the modern project management practice to make a time programme the Critical Path Method (CPM) is applied [2]. As an output of CPM the Project Schedule is created.

In the context of Delay Analysis, Finke (1999) describes three types of the Project Schedules[35]:

(i) *The As-Planned Schedule* defines a Contractor's original plan for performing its entire scope of work. It shows how and when the work would have been performed had there been no changes or delays.

(ii) *The As-built Schedule* reflects how a Contractor actually performed its work. This schedule also includes the impacts or effects of all changes and delays that were encountered during the course of the project.

(iii) *The Entitlement Schedule* according to Finke (1999) are meant to show when the project would have been completed had certain types of delays not occurred. It can represent an extended As-planned with certain classes of delays added or collapsed as-built schedule (i.e., the as-built schedule with certain classes of delays removed). In the present study all other schedules created for delay analysis purposes are also

[1] Time for completing the Works or section under Clause 8.2 "Time for Completion" in FIDIC Red Book 1999 [34].

[2] Other techniques and methodologies related to the Project scheduling are beyond the present study.

included in this group and are generalized that regardless to the applied DA technique an Entitlement schedule would be an as-planned or as-built schedule updated or improved to some extent.

2.1.3 Project Delay in Completion

If Planned Commencement Date is denoted as S^0 and Estimated Finish Date as E^0, then Time for Completion is the difference between these dates: $\theta^0 = E^0 - S^0$. Then the Actual Start Date is denoted as S^* (in the most cases $S^* \geq S^0$) and the Actual Finish Date as E^* with Actual Duration of the Project $\theta^* = E^* - S^*$.

Project Delay in Completion, as was defined above, is the difference between an original intent and a real project performance. More precisely is the positive difference between an estimated and actual *completion dates* , i.e. $D = (E^* - E^0)_+$.

2.1.4 Identification of Delay Events

Delay event is a unit of Delay Analysis. Same as a term 'delay', it is interpreted in different ways depending on a context or method of delay analysis. In the SCL Protocol (2002) delay event is explained as "an event or cause of delay, which may be either Employer Risk Event or a Contractor Risk Event" [90].

In a nutshell all delay analysis methods are about investigation of what delay events had occurred during the course of the Project, which of them had a critical impact on a Project completion date (i.e. became a reason, why the Project run late) and what party should be responsible for this event.

Delay Analysis generally starts with identification and forming the list of delay events. And what is understood by the term becomes a critical point in the discussion of particular technique implementation and liability.

Delay Event concept is revisited and extended in the Chapter 3 of the current Study.

s

2.2 Causation of Delay Events

There are several studies devoted to collecting and statistical analysis of the causes of delay events. Assaf et al. (1995) identified 56 causes of delays from previous studies[**?**]. Majid and McCaffer (1998) summarized 12 major causes of inexcusable delays and 25 factors contributing to them[70]. As Lovejoy(2004) summarized, the cause of a delay can be attributed either to (i) a specific party, (ii) a combination of parties, or (iii) unforeseeable and unalterable circumstances [67].

Delay is always associated with cost. As Love (2000) pointed out, when a project suffers a critical delay while substantial work is in progress, construction job site support costs such as trailers, supervision costs, maintenance, utilities, equipment and plants will continue to accumulate unless it is practical to mobilize these resources to another job site. Similarly, manufacturing resources idled by delay can cause continuing unforeseen costs [66].

Koehn et al. (1978) investigated the percentage of construction cost spent for delays by all consultants appearing on The ENR 500 Consultants Compilation due to governmental regulations. This study indicated that 30.3 percent of the overall yearly construction cost of projects in which The ENR 500 consultants are involved is spent for construction delays. In addition, the associated schedule delay due to governmental regulations was 29.9 months. [57]

In delay analysis practice all delay events are categorized in four groups by the responsible party and entitlement to the monetary compensation: excusable noncompensable , excusable compensable, nonexcusable and concurrent [86] [20]. This responsibility for delays on construction projects is often disputed and can itself become a subject of protracted litigation [105], but this issues are beyond the scope of the present Study. It is assumed, that responsibility for any particular delay event is known or can be identified.

Excusable Noncompensable Delays

Excusable noncompensable delays are delays that are not caused by the Owner or the Contractor and lies beyond the control of both parties. Contracts usually contain a clause called the force majeure clause, which enumerates the various causes of delays for which neither party is legally responsible. Force majeure delays most often entitle the contractor to an extension of time (EOT) or performance, but not to additional costs, although this depends on the contract language.[105]

Excusable Compensable Delays

Excusable compensable delays can be delays that are Owner's responsibility and that result in both a time extension and compensation to the contractor.[105]

Nonexcusable Delays

Nonexcusable delays can be attributed to the actions, or inactions, of the Contractor. When a Contractor causes delays to the completion of a project, such delays preclude the contractor from obtaining a time extension and may also trigger delay damages against the contractor [105].

2.3 Complexity in Delay Analysis

Two main reasons, why delay analysis bring debate in the society, are the complex nature of delay and fundamental problems with reliance on existing delay analysis technology, substantially a high level of the subjectivity involved in its theory. Whatever the method or approach is, the analyst always has to make judgments about a wide range of issues that affect the analysis and its results.[33]

2.3.1 Non-additive feature of Delay

Delays have been proven to show a non-additive feature, which means that durations of two or more delay events does not equal to the sum of durations of these delay events (2.1).

$$\omega^1 + ... + \omega^k \neq \sum_{k=1}^{K} \omega^k \tag{2.1}$$

This happens because delays tend to occur in parallel and can be distinguish in terms of its criticality. Moreover, delay in an activity may not result in the same amount of project delay. A delay caused by a party may or may not cause damage to another party [7]. Simply $\omega^k \neq D^k$, where ω^k is a delay event, and D^k is delay to completion, caused by this event.

2.3.2 Global and Net Impact Delay Analysis Methods

On the earlier stages of delay analysis development, when the non-additive feature was not yet established, two methods were proposed - Global and Net Impact.

In the global method the excusable delay periods were simply added to the end of the planned completion date, and then the actual completion date was compared with a calculated date. If the latter was equal or later than actual completion date, the contractor was entitled to full extension of time (2.2).

$$E^0 + \Omega_E \geq E^*, \Omega_E \in \Omega \tag{2.2}$$

E^0 and E^* - planned and actual finish dates,

Ω - set of all delay events, occurred during the Project,

Ω_E - subset of excusable (Owne-caused) delay events.

According to Farrow (2007) in the English case of McAlpine Humberoak versus McDermott International (No. l) (1992) 58 BLR 1, the claimant sought to demonstrate its delay case by arguing that every day worked on particular variations equated to a days delay to the project and the argument was not accepted. [33]

The *Net impact method* is essentially the same as the global impact method but with the refinement that the issue of concurrency of delays is considered.[33] If two or more listed events happened at the same time, only the longest one is considered. Calculations procedure is the same as for Global Impact Method.

The global and net impact approach is quick and simple but never contractually supportable and provides no cause and effect. It ignores other delays occurring at the same time and does not consider timing, concurrency, or dominance of delays. It also ignores any actual delays caused by the contractor. This method has also been repeatedly criticized by the courts because it fails to consider the fundamental issue of criticality (that is, whether the works were delayed or not) and ignores reality. [33]

Courts and arbitration panels resist the use of the global impact technique because it wrongly assumes that every delay has an equal impact on the project duration. [7]

2.3.3 'Who owns the Float?'

Total float is the time difference between the earliest finish and the latest finish of an activity [85]. All activities on the same path co-share the total float in that path (Callahan et al., 1992). It is a by product of the CPM analysis (de la Garza et al., 1991). Free float presents the amount of time that an activity can be delayed without delaying the earliest start of its following activities. Raz and Marshall (1996) mention two other types of float, namely interfering float and independent float. The first refers to the difference between total float and free float while the second refers to the difference between the interval of time from the latest finish of an activitys predecessors to the earliest start of its successors, and the activity duration. Float is an important measure of schedule flexibility associated with activities and an indicator of the amount to which the schedule can absorb delays without affecting the project duration (Raz and Marshall, 1996).

Ownership of the float is an issue constantly addressed in various studies, and yet no decision was found on how to allocate it among the parties.

The SCL Delay and Disruption Protocol's position is that if the clause stating the entitlement of float is not specified in the contract, float should belong to a project [90]. In other words, the use of float is govern by the principle - "first-come first-served". On the other hand, Scott et al.'s (2004) survey of 46 UK professionals employed by owners, contractors, and claims consultants shows that the majority of respondents believed that the contractor should have exclusive control of float while only few owners supported the position that float should be allocated on a first-come first-served basis [91]. de la Garza et al. (2007) agree with the British professionals that float is exclusively for the benefit of the contractor and add that float should be traded as a commodity, and the Contractor is entitled to sell the float in case the owner needs to consume such float.[37]

Householder and Rutland (1990) suggest that the use of float should be reserved for the party who loses or gains as a result of fluctuation in the project cost [46]. Ponce de Leon (1987) suggests a compromise position regardless of the type of contract: allocating float in a shared way. [85] Activities would be allocated a percentage of the float available to the path based on the individual activity's duration. If an activity is delayed beyond its allocated float, then time extension may be justified to preserve the allocated float of other activities in the approved schedule. Another compromise solution is offered by Pasiphol and Popescu (1994) who propose a qualitative method to distribute total float into each activity prior to executing a project.[81]

In relation with delay analysis the problem with float ownership lies in two points of argument: (i) float ownership may be not considered in delay analysis technique at all or (ii) its distribution method may cause debate.

2.3.4 Concurrent Delay Situations

According to Peters (2003) there is no consistent agreement on what concurrent delay actually means [83], but in general concurrent delay can be described as two or more delay events that occur at the same time, either of which would cause a delay but if either of them had not occurred, the project schedule would have been delayed by the other. [27, 50, 86, 96, 82].

When concurrent delay exists, the assessment of delay damages and/or time extensions is difficult and often result in serious disagreements [7]. Both owners and contractors employ concurrent delays as a strong defense tool against each other. For instance, owners use them to protect their interest in obtaining liquidated damages, while contractors use them to neutralize or waive their inexcusable delays and hence avoid damage entitlement [19].

Nguyen (2007) has summarized the inconsistency in allocation of responsibility when concurrent delay occurs (Table 2.1) [78].

2.3.5 Subjectivity in Delay Analysis

Delay analysis methodologies provide a set of rules for examining project delay, but these many subjective assessments work against the intent of a logic-based empirical analysis of a project and they undermine the analytical or clinical nature of the process. In addition, the rules of the methodologies can be ill defined or require judgment in applying them, and this again increases the level of subjectivity. [33]

For this reason, the answer that a methodology provides is only as good as the accuracy of the available information, the assumptions inherent in the methodology, and the reasonableness of the subjective decisions made by the analyst.[33]

No	Literature	Concurrent delays		
		Excusable & Inexcusable	Excusable & Compensable	Compensable & Inexcusable
1	Ponce de Leon (1987)	Excusable	Compensable	Excusable
2	Reams (1989); Battikha and Alkass (1994)	Excusable	Excusable	Not available
3	Arditi and Robinson (1995); Al-Saggaf (1998)	Inexcusable	Excusable	Not available
4	Rubin (1983); Galloway and Nielsen, (1990); Wiezel (1992); Alkass et al. (1995); Schumacher (1995); Galloway et al. (1997); Kartam (1999); Stumpf (2000); Reynolds and Revay (2001); Niesse (2004)	Excusable	Excusable	Excusable
5	Construction (1993); Baram (2000); Construction (2002)	Inexcusable	Excusable	Inexcusable
6	Kraiem and Diekmann, (1987); James (1991); Kutil and Ness (1997); Finke (1999); Ness (2000); Bubshait and Cunningham (2004)	Excusable	Excusable	Excusable or Apportioning
7	Hughes and Ulwelling (1992); Wickwire et al. (2003)	Excusable	Excusable	Apportioning

Table 2.1: Divergent and inconsistent perspectives on concurrent delays

Chapter 3

CONVENTIONAL DELAY ANALYSIS METHODS

3.1 General Formulation of Conventional Delay Analysis Methods

There are various papers devoted to description of delay analysis methods and assessing its strong and weak sides [2, 4, 7, 22, 25, 33, 51, 56, 73]. However even referring to a method as commonly used or conventional each author provides his own interpretation of what particular technique is. Sometimes one interpretation is so different in details from the other that in fact can be considered as another method.

In order to cover most of existing DAMs and not to get into confusion with names and interpretations, all techniques in the following literature review are considered within certain groups of methods rather than one by one, with exception of several relatively new methods.

3.1.1 As-Planned Impacted group of methods

This group of DAMs is united by application of the concept of adding delay events to the as-planned programme and then assessing its impact on the project completion date.

3.1.1.1 Review of existing techniques

Some examples of these techniques described in the literature:

Impacted As-Planned Technique:

This method starts by sorting the delay events in a time sequence manner. Also the critical path(s) of the as-planned schedule are identified. Then the critical path is placed on a time sequence with sorted delayed events. The difference between each impacted as-planned critical path(s) and the previous one is the delay resulting from the delayed event. Compensable damages will be summation of all critical path delayed events of owner-caused-delay. On the other hand, the summation of all critical path delayed events of contractor caused delay will be liquidated damages. [99]

But-for or collapsing technique:

One party taking the as-planned schedule and injecting all delays they are willing to accept responsibility for. The updated schedule yields a revised project completion date and is compared against the as-built schedule. The difference between the as-built and the revised project completion date is a result of delays that were beyond the claimant control. For example, if the contractor were using this technique they would identify and include only non-excusable (contractor's fault) delays into the as-planned schedule. As a result an adjusted schedule would be generated with a revised completion date. The difference between this adjusted completion date and the as-built completion date is due to delays that are the owner's fault. [4]

What if method:

The what-if method adopts the anticipated or as-planned schedule as its baseline. First delay attributable to one side is added to the as-planned schedule and the impact on project duration due to those delays is determined. Then this process is repeated for the delays attributable to the other side. And finally, the amount of delay in total project duration attributable to each side is determined.[89]

As-Planned Schedule Delay Analysis:

The As-Planned Method measures the contractor's planned or intended. Under [this method] the scope of changed work is reviewed to determine where and how the revisions should be incorporated into the schedule. As-Planned Method requires all information concerning all delays to be analyzed at one time rather than making separate calculations for each delay. The effect of the changes on the schedule is determined by comparison of the schedules before and after the changes were incorporated into the schedule. [22]

As-Planned CPM Technique:

Various delays, disruptions and suspensions are presented as events with duration and added to the baseline schedule including appropriate sequence. The baseline schedule is allowed to calculate with the delaying events and the claimant's time extension is the duration between the planned completion and the new impacted planned completion.[84]

But-for Analysis using As-Planned CPM:

Uses same premise as "As-Planned CPM Technique" but chooses events for which the claimant is willing to accept responsibility and inserts the delaying events into a baseline schedule. The but-for technique adjusts the as-planned completion date for delays that are clearly the claimants responsibility and assumes that the differences between the adjusted as-planned and actual completions are responsibility of other party, forming the basis of the schedule delay.[84]

As-Planned Technique:

It impacts the as-planned schedule with owner-caused delay. Then it compares the affect of the as-planned schedule before and after the impact to determine the compensable damages. Liquidated damages can be calculated by impacting the as-planned schedule by all contractor caused delay events. Then, compare the project completion date between the two schedules. This technique is implemented by two methods. First method uses a gross measure, which impacts the as-planned schedule of owner-caused delay or contractor-caused delay as one shot. The second method uses a unit of measure which impacts each individual delay event on the as-planned schedule to determine each of their effects.[21]

Impacted As-Planned Analysis:

In this technique the scheduler takes the as-planned schedule and adds new activities that represent delays (generally caused by the other party) to demonstrate why the project was completed later than planned. The selection of what new activities are added to the schedule depends upon who is performing schedule analysis.[107]

As-planned impact method:

The as-planned impact method analyzes the potential effect of delay events by impacting them onto the original baseline i.e., on the original planned program and projecting the completion date using the original sequence, logic, and timing of remaining activities. This method can be used to show the potential delaying effect of the employer's delays, or of the contractors delays, or of both together. [33]

As-planned but for:

The analyst impacts the planned baseline program with the assessed implications of the events a party considers it is responsible for and the combined influences of these are analyzed. The impacted completion date is then compared with the as-built completion date (that is, when the project was actually completed) and the difference is said to be how much earlier the project could have finished but for all the other delay events (imposed by the other party) but which have not been analyzed. The period between the analyzed date and the actual completion date is said to represent either the contractors entitlement to an extension of time or the employer's entitlement to deduct liquidated damages, depending on which set of events have been analyzed.[33]

Impacted as-planned:

The delay events that the analyst wishes to model are introduced into the baseline schedule, and linked in an appropriate manner. Having incorporated the delay events, the schedule is then modeled or recalculated. Any resultant impact upon completion is determined to be critical delay impact of introduced delay event(s). The impacted as-planned analysis method lacks balance and fairness because it typically includes only one party's delay events, while at the same time assuming perfection from the other party. [11]

Modeled / Additive / Single Base (MIP 3.6):

The simulation consists of the insertion or addition of activities representing delays or changes into a network analysis model representing a plan to determine the impact of those inserted activities to the network.[1]

3.1.1.2 Variations of API methods

Despite using the same basic idea, all methods within this group can be distinct in details and algorithms. The variations of the API methods can be recogniaed:

By the Party responsible for delay events

Tony Farrow (2007) in his as-planned impacted technique description points out, that it can be used "to show the potential delaying effect of the employer's delays, or of the contractor's delays, or of both together". On the other hand D.Barry (2009) interprets impacted as-planned as a method, which "typically includes only one party's delay events, while at the same time assuming perfection from other party". In the same paper T.Farrow describes method called as-planned but-for, which indeed uses only one party delay events for analysis.

Same way Pinell (1998) distinguish As-Planned CPM Technique and "But-for" Analysis using As-Planned CPM, saying that the former presents as events "various delays, disruptions and suspensions" (attributable to different parties), and the latter "uses same premise as "As-Planned CPM Technique" but chooses events for which the claimant is willing to accept responsibility" [84].

Also there are methods which use combination of both approaches - addition of delay events from one party and then from the other. For example, what-if method described by Schumacher (2005), according to whom "first delay attributable to one side is added to the as-planned schedule and the impact on project duration due to those delays is determined. Then this process is repeated for the delays attributable to the other side".

In some cases the author does not specify it in the description of technique, and it becomes itself a matter of interpretation.

By the grouping of delay events

All methods within this group are united by the "adding to as-planned" principle, however delay events may be inserted into baseline schedule at least in two different ways:

- All at the same time, as a "single shot"
- Or one-by-one.

As-Planned Method, described by Bubshait et.al (1998) "requires all information concerning all delays to be analyzed at one time rather than making separate calculations for each delay". [22] But, for example, Farrow (2007) suggest while using as-planned impacted method not only to assess "likely critical delaying effect of *each* delay event", but also "impact the individual events onto the program in *chronological order*".

Bramble et.al (2000) also point out that As-Planned Technique "is implemented by two methods. First method uses a gross measure, which impacts the as-planned schedule of owner-caused delay or contractor-caused delay as one shot. The second method uses a unit of measure which impacts each individual delay event on the as-planned schedule to determine each of their effects" [21].

By criticality of analyzed delay events

After decision on which party's delay events should be analyzed and in which order, the Delay analyst should also make a decision whether only events laying on the critical path would be used or all other events as well. It is generally believed that only delays within a

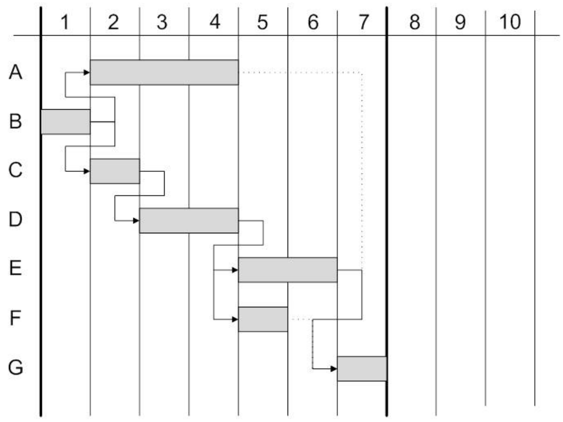

Figure 3.1: The Planned Schedule

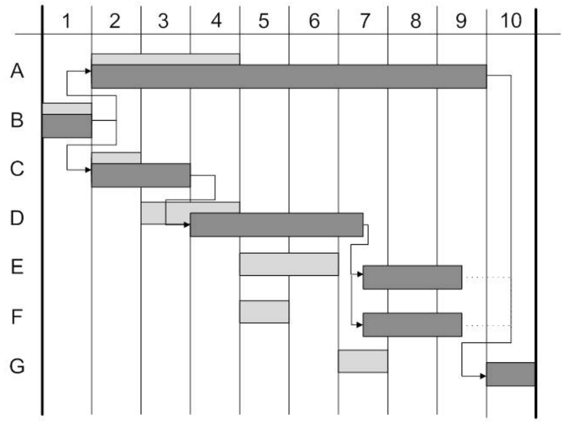

Figure 3.2: The Actual Schedule

critical path have impact on project completion date, however since the project schedule is constantly changing, activities gain and loose float, what set of delay events is chosen from the beginning may define the analysis results.

In as-planned group of methods some authors especially emphasize and distinguish methods which require using critical path only or a full as-planned schedule. For example, one of the differences between As-Planned and As-Planned Impacted Technique, described by Mohan et.al (2006) is that the former incorporates into schedule all delay events, and the latter first determined the critical path and injects delay event to as-planned critical path.

3.1.1.3 Application of API to a Simple Case

To review and compare the As-Planned Impacted group with other group of methods we will be using a simple case example. Its as-planned program, the original plan or baseline is shown in Figure 3.1 Project consists of seven activities A, B, C, D, E, F, G. The start day is at the beginning of the first week, the project completion date end of 7th week. Thus overall planned duration of the project is 7 weeks.

There are three paths in this project: (a) B-A, total length = 4 weeks; (b) B-C-D-E-G, total length = 7 weeks; (c) B-C-D-F-G, total length = 6 weeks.

The Critical path is the longest path of the project, i.e. B-C-D-E-G. All activities on the critical path have zero total float, and delay on them will cause delay to the project completion date. Activities A and F do not belong to critical path and have total float respectively 2 and 1 weeks. Thus they can be delayed up to these periods without causing delay to the whole project.

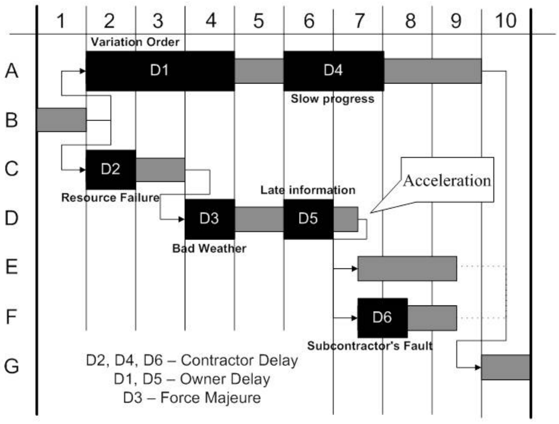

Figure 3.3: The Delay Events and Acceleration.

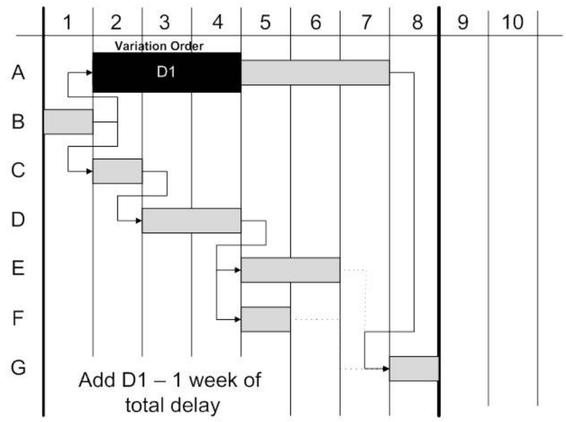

Figure 3.4: Impact of D1 on as-planned schedule

Figure 3.2 identifies the as-built program of this project compared with original baseline. The critical path has changed, as it usually happens, and now includes activities B-A-G. The actual duration of the project was 10 weeks. Comparison to as-built program shows that there was s 3 weeks delay.

During investigation phase all delay events have been identified and noted in Figure 3.3 Delay event D1 is variations issued by the Owner and required additional work; D2 resource failure, which is the Contractor's responsibility; D3 severe weather conditions (Force Majeure); D4 slow progress by the Contractor; D5 late information from the Owner; D6 delay caused by Subcontractor's mistake.

Delay Events D1, D3 and D5 considered being excusable, and for them the Contractor normally would be granted the extension of time. While D2, D4, D6 are non-excusable, and in fact the Contractor may be obligated to pay liquidated damages to the Owner. D6 is a good example of noncritical delay, which had no influence on project completion date because of 1 week planned float on activity F. Also the investigation shows that there were acceleration measures on activity D taken by the Contractor.

In the following example we will demonstrate two modifications of the As-Planned Impacted methods: First, where delay event(s) are added into the As-Planned program in chronological order and then the project completion date is reanalyzed until all of the delays have been impacted. And second is where the analyst chooses set of delay events related to only one party and then adds them into the planned baseline program. The impacted completion date in such case is then compared with the actual completion date, and the difference is said to be how much earlier the project could have finished but for all

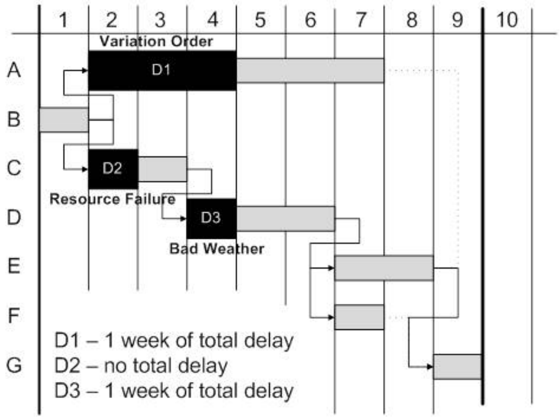

Figure 3.5: Impacts of D1, D2 and D3.

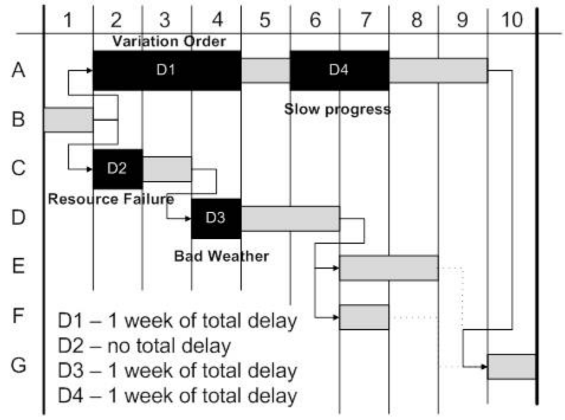

Figure 3.6: Impact of D4

other events (related to the other party) which have not been analyzed.

3.1.1.3.1 Straight logic

In the example case the as-planned program (Fig.3.1) is impacted with the delay event D1, which is variations on activity A, and the project end date is recalculated, as shown on Fig.3.4. Because of the 2 weeks total float on Activity A the 3-weeks delay caused only 1 week delay to project completion. The critical path now runs through activities B-A-G.

Then D2 is added to the as-planned program. Since the activity C is no longer on the critical path and has 1 week of total float, the 1-week delay will not cause delay to completion. The next event D3 will delay the project for 1 week, as shown on Fig.3.5 The critical path changes back to B-C-D-E-G.

Next event D4 added to the program causes 1 week of critical delay. Critical path changes again to B-A-G, as can be seen on Fig.3.6. Further delays D5 and D6 do not change the completion date.

Therefore, the API method allocated the total delay of 3 weeks between the parties: the Contractor should receive 2 weeks extension of time and the Owner should receive 1 week of liquidated damages.

It is not difficult to notice that if delay event are added to as-planned program in different order, for example first D2 and then D1, the analysis shows that other events had critical influence to project completion and hence the results will be different.

That is: the Contractor should receive 1 week extension of time (for D3 bad weather) and the Owner should receive 2 weeks of liquidated damages (for D2 and D4 resource failure and slow progress by the Contractor).

3.1.1.3.2 But-for logic

In this case the contractor-caused critical delays (D2, D4) are analyzed. First the D2 is added to as-planned program, the impacted completion date now is week 8 (1 week of total delay), then the D4 delay event is added, but because of the 2 weeks float on activity A it will not cause delay to completion. The conclusion is that but for all other delays, the contractor would have finished on week 8. Hence the contractor should receive 2 weeks extension of time and compensate 1 week of liquidated damages to the owner.

3.1.1.4 Advantages of As-Planned Impacted Methods

There are not so many advantages recognized in application of these techniques. Farrow (2007) points out that the "the strength of this method is that the process avoids the need to analyze actual progress records in detail because the key elements of the methodology are the original baseline program and a schedule of delay events... It is reasonably quick, as there is no need to consider actual progress of works or the timing of events".[33] "The impacted as-planned analysis method enjoys the benefit of simplicity, and will therefore be inexpensive to prepare" [11].

3.1.1.5 Critisizm of As-Planned Impacted Methods

There are four main points of criticism toward the As-Planned Group of methods:

1. The As-Planned program may not reflect what actually happened in the project.

   "This method ignores updating the schedule with as-built data. This might lead to failure to depict the actual situation of delay events" [99].

   "The technique does not consider the as-built schedule, status of schedule when delays occurred, and assumes that the claimant is responsible for concurrent delay. The fatal flaw of this technique is that actual sequence and progress may have been significantly different from the planned schedule" [84].

   Farrow (2007) agrees, that "there are two significant weaknesses of the method. First, the original baseline program may not be realistic model on which to base the whole analysis (because the works were probably carried out in a different sequence and at a different time from that originally planned). Second, since actual progress is not considered, this method does not demonstrate what actually caused delay to the works" [33].

"The timing of delay event is just not considered: it is possible that at the point in time when the delay event occurred it was unlikely to have any impact upon project completion due to progress or events elsewhere." [11]

2. Other point is related to those types of techniques which use only one party's delay events. "One of the negative sides of the as-planned technique is the fact that it ignores the affect of other party delay events and excusable, non-compensable delay events" [73].

   If the method does not require to apply all delay events, but first the analyst has to choose only critical or caused by particular party, another drawback can occur. As Zack (2001) showed, "the selection of what new activities [representing delays] are added to the schedule depends upon who is performing the schedule analysis" [107].

   The impacted as-planned analysis method lacks balance and fairness because it typically includes only one partys delay events, while at the same time assuming perfection from the other party [11];

3. If delay events are added not in a single shot then the order may start plying a critical point. As Farrow stressed, "if delay events are added to the planned program in a different order, the analyst will draw a different conclusion as to which events caused the completion delay and what extension of time should be awarded" [33].

4. Finally, this method does not define exactly the concurrent delay [73].

   Putting it in a short form, As-Planned group of methods are a "theoretical investigation" [33].

3.1.2 As-Built Collapsed Group of Methods

Techniques fallen into this group use the idea of subtracting delays from as-build schedule (i.e. collapsing it).

As-built but for

The as-built program is first constructed from the contractors progress records and linked together into a critical path network. This becomes the model to be analyzed. A schedule of delaying events is created, including a measure of their impact (for example, a start delay, a finish delay, or an activity prolongation). A delay event lying on the critical path is removed from the program and the model is reanalyzed. The difference in the overall program duration before and after this removal is said to represent the period of critical delay caused by the particular delay event removed. This process is then continued until all delays have been removed and the model has been fully collapsed (the method is also referred to as the collapsed as-built method). [33]

Collapsed as-built or but-for analysis

Assuming that the logic between activities can be developed appropriately, the next task is for the analyst to identify the incidence of delay within the activities on the as-built schedule. The analyst then ensures that the delay aspect of the relevant activity can be removed in order to stimulate what as-built schedule would have looked like if the delay event had not in fact occurred. [11]

But-for method

The but-for method subtracts the delay attributable to the owner from as-built schedule. The compensable delay is the difference in duration between the as-built schedule and but-for schedule. A but-for analysis based on on subtracting delay attributable to the contractor from as-built schedule will nearly always result in an incompatible result [89]

Collapsed As-built Method

The method purports to address concurrent delays by removing delays from the as-built schedule and collapsing it. Durations of delays are usually arbitrary established; a process often manipulated to cover up the effect of claimants delay and does not consider changes to the critical path. [84]

Impacted As-Built CPM

Technique is similar to collapsed as-built method except that the as-built schedule progress is removed and the as-built schedule is developed in CPM format. Delaying events are identified and depicted in the as-built schedule as distinct activities, which are then tied to specific work activities by constraints. The critical path os determined twice, once in the as-planned schedule and again at the end of the project. Claimant requests schedule extension for time between the as-planned completion and as-built completion or the impacted as-built CPM completion. If the impacted completion date is earlier that the planned completion date, the claimant will claim that it is a result of constructive acceleration. The critical path is calculated only twice. [84]

But-for Schedules

This technique attempts to create an as-built schedule, identify actual delays caused by two parties, and then reove on partys delays from the as-built schedule to collapse the schedule leaving in the schedule those delays caused solely by the other party. The argument is, but-for the other partys delays, this is when the project would have been completed. The amount of delay and the resulting damage are then calculated. [107]

Modified But-For Method

The process starts by indentifying all the daily delay events (o, c, and n) on the as-built schedule. Since three parties are involved (contractor, owner, and third party), there can be seven mathematical combinations of these events, as follows: o, c, n, o+c, c+n, and o+c+n. The MBF method then removes these seven combinations, one at a time, from the as-built schedule, and the resulting project duration is used to calculate the values in the Venn diagram. This Venn diagram shows a representation of all types of critical delay combinations. [72]

But-for or collapsing technique

This technique is implemented twice, once from the owner point of view and the other from contractors point of view.

The but-for schedule from the owners point starts with taking off all the contractors delay events from as-built schedule, and comparing the as-built schedule with this collapsed schedule. The difference will represent the contractors delays for liquidated damages.

A but-for schedule from the contractors point of view is the opposite of the owners but-for schedule. It starts with taking off all the owner delay events from the as-built schedule with the collapsed schedule. The difference will represent the owner delays, for which the owner is liable to pay damages to the contractor.

There are two implementation methods for this technique, gross measure and unit of measure. The unit of measure is different from gross measure in that rather than collapsing the as-built schedule with the delayed events of one party, it collapses each delayed event individually. [73]

Modeled/ Subtractive/ Single Simulation (MIP 3.8)

This is a modeled technique relying on a stimulation of a scenario based on a CPM model. The simulation consists of the extraction or subtraction of activities representing delays or changes from a network analysis model representing the as-built schedule to determine the impact of those subtracted activities to the network.[1]

Modeled/ Subtractive/ Multiple Base (MIP 3.9)

The simulation consists of the extraction of entire activities or a portion of the as-built durations representing delays or changes from a network analysis model representing the as-built condition of the schedule to determine the impact of those extracted activities to each network model.

The subtractive simulation is performed on periodic network analysis models representing intervals of the as-built schedule. Each model creates a time period of analysis that confines the quantification of delay impact. Forecasted delays beyond an analysis period, however, may also need to be extracted at the time that the forecasted delays are introduced into the schedule.[1]

3.1.2.1 Variations of CollAB methods

As Farrow (2007) pointed out, collapsed as-built methods can be applied by removing all the excusable events, either singularly or together, or by grouping similar excusable events together and removing each group individually. Whatever the approach, each invariably produces a different answer.[33]

The main variations of CollAB methods can by recognized:

- *By the Party responsible for delay events*

  Arditi et.al (2006) states that when a contractor conducts a collapsed as-built analysis, the analyst considers only the delays caused by the owner to prove the effects of owner-caused delays on the project completion and does not include contractor-caused delays in the analysis [7]. One party delay events are also considered in the But-for Schedules method, decribed by Zack (2001), while in the Modified But-For Method (by Mbabazi et.al (2005)) all delay events (attributable to all parties) should be considered during analysis.

- *By the grouping of delay events*

  Same as with API methods delay events can be removed one-by-one or in a 'single shot'.

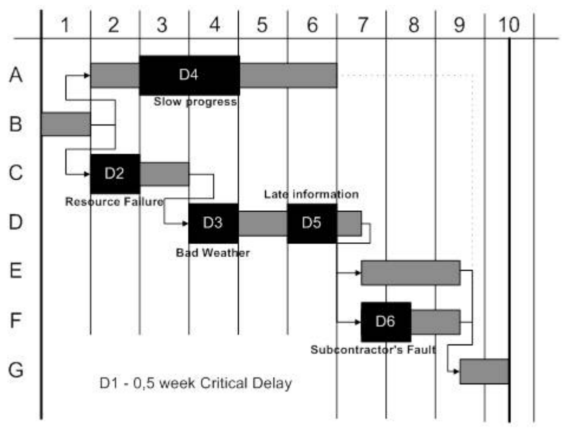

Figure 3.7: As-built program after D1 is removed

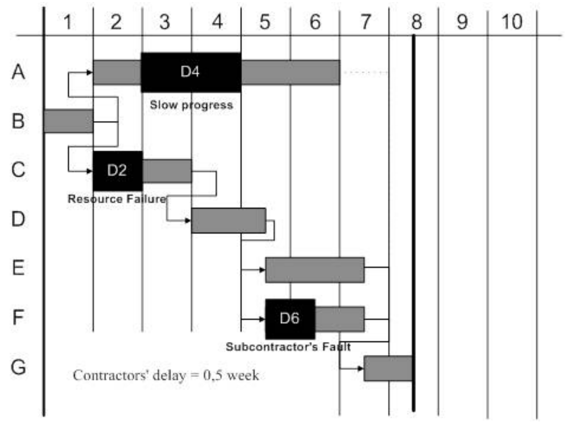

Figure 3.8: As-built program after all excusable delays removed

- *By criticality of analyzed delay events*

Apllying only delay events belong to identified as-built critical path or all delay events.

3.1.2.2 Application of CollAB methos to a Simple Case

We will be using same example as we used for the As-Planned Impacted methods and will demontrate two modifications of As-built Collapsed methods: Fisrt, where delay event(s) are subratced from the As-Built program in forward chronological order and then the project completion date is reanalyzed until all of the delays have been subtacted. The difference in overall program duration before and after this subtraction is said to represent the period of critical delay by the particular delay event removed.

And second is where the analyst chooses set of delay events related to only one party and then subracts them from the actual program. The "collapsed" completion date in such case is then compared with the actual completion date, and the difference is said to be how much earlier the project could have finished but for all other events (related to the other party) which have not been analyzed.

3.1.2.2.1 Straight logic

In the example case, the first employer-caused event, the 3-weeks variation delay D1, removed from as-built program (Fig.3.7). The new completion date is becoming in the middle of week 10, which means that but for the D1the project could have been completed 0,5 weeks earlier. Thus D1 has caused 0,5 weeks of critical delay.

Continuously removing other delay events the analyst can draw the following conclusions: D2 1 week of critical delay; D3 1 week of critical delay; D4 no critical delay; D5 0,5 week of critical delay.

This means that the Contractor should receive 2 weeks of extension of time and compensate 1 week of liquidated damages to the Owner.

In is easy to notice that if the event were removed in different order, the allocation of total delay among the parties would be different. For example, if D2 would be removed first and then D1, the analyst would conclude that the Contractor should receive full extension of time.

3.1.2.2.2 But-for logic

In the example case the excusable delay events are subtracted from the as-built schedule (Fig.3.8). As a result, the Contractor should receive 2,5 weeks of extension of time and compensate 0,5 week of liquidated damages.

The ABBF method used to analyze non-excusable delays leads to different conclusions. The Contractor should receive 1,5 of EOT and compensate 1,5 weeks of LD to the Owner.

3.1.2.3 Advantages of As-Built Collapsed Methods

The as-built schedule depicts the factual information concerning the work that has been undertaken [7]. Courts and boards in the US consider the collapsed as-built method to be useful because the activities in this method are consistent with actual occurrences on the project [93].

The collapsed as-built method is often selected when reliable schedules cannot be readily obtained from project records or the project does not have scheduling information[7]. According to Lovejoy (2004), the collapsed as-built analysis is the most practical approach since it offers a good combination of benefits [67].

3.1.2.4 Critisizm of the As-Built Collapsed Methods

While this method is comfortably based upon the facts, it should be kept in mind that the conclusion reached is a hypothesis and not a fact. [11] Whatever the approach, each invariably produces a different answer! [33]

There are four main points of criticism toward the As-Built Collapsed Group of methods:

1. The obtained As-Built program may not reflect what actually happened in the project.

Although the principles of this methodology are straightforward, its application is not. It is dependent on the creation of logic links that may not have been set down during the project and which were never agreed on in any contemporaneous program. Also, removing delay events (and logic links), retrospectively does not reflect the actual way the works may have been progressed.[33]

The key difficulty in this method of analysis is in developing retrospectively the logical relationships between as-built activities. [11]

2. Subtraction of delay events related to only one party

   But-for analysis based on subtracting delay attributable to the contractor from the as-built schedule will nearly always result in an incompatible result.[56]

3. Does not consider changes in critical path

   The technique is flawed to the point that the critical path calculation is somewhat contrived since the calculation is after the fact and is calculated only twice[84]

   A process often manipulated to cover up the effect of a claimants delay and does not consider changes to the critical path[84]

4. Concurrent delay is not addressed.

   Delay events are collapsed in one shot to the as-built schedule; and any changes in the critical paths during the course of the delay event are not considered. This has the potential of giving inaccurate results. Concurrent delay is not addressed in an accurate way. [73]

   This method can only be used satisfactory in very discrete circumstances, such as where the key delay being addressed is very close to the end of the project, or where the project is wholly linear in nature.[11]

3.1.3 Contemporaneous Period Analysis group of methods

3.1.3.1 Review of existing techniques

Some examples of these techniques described in the literature:

Windows Analysis:

The technique is to validate the as-planned or baseline schedule, and then, using contemporaneous project documentation, update the schedule one period at a time (monthly, quarterly, seasonally, etc.). The technique builds one period analysis upon the previous period's analysis, examining each new period for delay, causation and liability as the analysis proceeds [107].

Window/snapshot method:

The as-planned and as-built programs are first established, and then logically linked so that they become critical path programs. The as-planned program becomes the model. The overall project duration is then divided into periods in order to make the analysis more incremental [33].

In practice, the increments coincide with the frequency of the contractor's progress reporting cycle or with major milestone events, such as completion of foundations, completion of roof, etc. For each time period or "window" the duration, progress, and logic of the work actually carried out in that month are imposed onto the as-planned model. If any major planned program revisions are made in that period, these are also introduced into the model. The model is then time analyzed. The end date is the result of actual progress achieved to date (i.e., up to the end of the window) and planned durations and logic for the remaining work [33].

The difference between the end date from this analysis, and the end date from the previous window or snap-shot, represents the period of delay or mitigation which arose in the current window [33].

Snapshot analysis method:

The baseline schedule is updated at regular intervals, usually monthly. Each update provides a 'snapshot' of status at that point in time, and from which one can discern two important pieces of evidence. Firstly, the sequence of activities which represents the critical path to completion at that date, and secondly, the extent of actual delay incurred to completion as at that date.[11]

Contemporaneous Period Analysis (CPA):

The CPA method breaks the construction period into discrete time increments and examines the effect of the delay attributable to each of the project participants as the delays occur. It adopts the as-planned schedule as its baseline, but the as-planned schedule is periodically updated at the end of each planned time increment, and the updated schedule becomes the baseline for the ensuring time increment. This process is continued through to the project completion.[89]

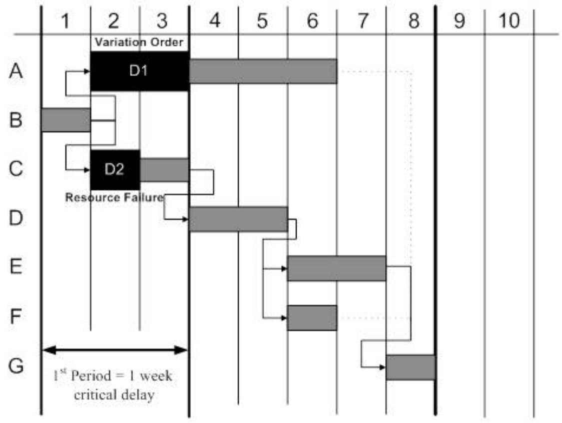

Figure 3.9: CPA method, First Period update.

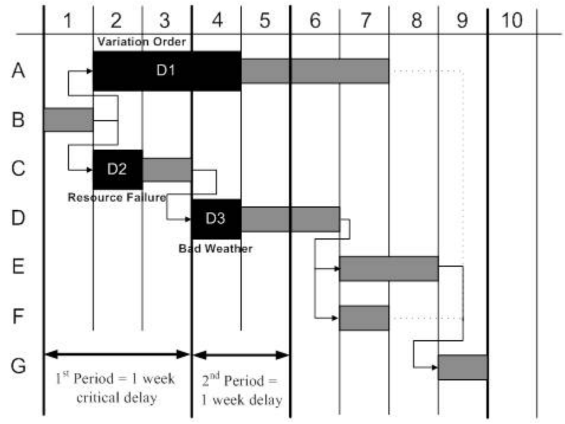

Figure 3.10: CPA, Second Period Update

Traditional Windows Analysis (TWA):

TWA method divides total project duration into digestible time periods (windows) and analyzes the delays that occured in each window successively [42]

As-Planned versus as-built windows analysis method:

Windows are used to break the project into manageable sections or periods. The as-planned versus as-built windows method seeks to first locate and identify the project's actual critical path. Having established the actual critical path from start through completion, it can be compared to the corresponding planned activities, and from this the incidence and extent of delay can be adduces [11].

3.1.3.2 Advantages of CPA Group of Methods

This method puts delays in their proper time frame within the overall context of the project [107] and can be very effective and reliable, provided that the snapshot schedules accurately and reasonably reflect the status of works on the respective dates [11].

3.1.3.3 Criticism of CPA Group Methods

Windows-based analysis methods are computationally intensive and usually produce different results according to the window size [51].

3.1.3.4 Application of CPA methods to a Simple Case

Same example used for the As-Planned Impacted and Collapsed As-Built group of methods will be used to demonstrate application of CPA method.

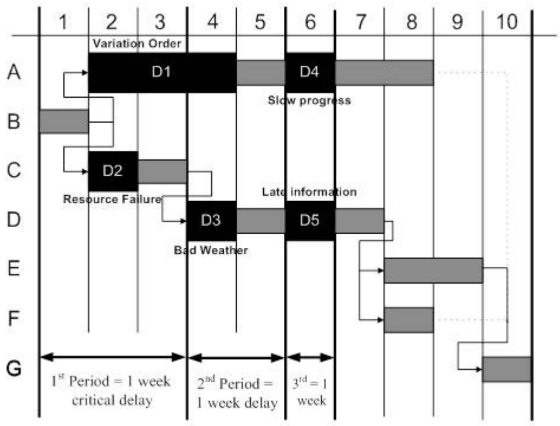

Figure 3.11: CPA, Third Period update

The overall project duration is divided into periods (or windows). Then the analyst updates the as-planned program (Fig. 3.1) within the first window using as-built information including all the delays and acceleration measures in that period.

The remaining as-planned schedule is maintained beyond this window. The period of delay or mitigation arose in current window is represented by the difference between the project completion date after updating this window and the project completion date from the previous window.

In simple example case the time period chosen is 3 weeks. After the First period update, the project completion date becomes week 8 (see (Fig. 3.9)

The Critical Path runs through activities B-C, hence the 1 week of critical delay is caused by delay event D2. Next period update is 2 weeks (Fig.3.10) shows that delay event D3 on critical activity D has caused 1 week of critical delay.

Third period update is 1 week (Fig.3.11) and shows that the delay event D5 occurred on crit-ical activity D and caused 1 week of total delay to the project.

As a conclusion the contractor should receive 2 weeks of EOT (for bad weather conditions and late information from the Owner), and the Owner should get 1 week of LD (for failure of the Contractors resource).

3.1.4 Time Impact Analysis Group of methods

In SCL Protocol Time Impact Analysis is considered to be “the most thorough method of analysis” and is the preferred technique to resolve complex disputes related to delay and compensation for that delay [90]. However, as it happens to all conventional delay analysis methods, the illustrations and interpretations of this technique varies depending on the author or the analyst.

Some examples of how Time Impact Analysis is interpreted in the literature could be found below.

3.1.4.1 Review of existing techniques

Impact/update method

This method applies events to the as-planned model on a month by month, or window by window, basis to derive the monthly entitlement and then goes on to consider the actual monthly production, i.e., to contrast the contractor's actual performance, in terms of culpable delay or mitigation.

For each of the window periods, the delay events that are alleged to have arisen in the period in question and are employer-caused items are impacted on the planned model and time analyzed. The resultant project end date will reflect the notional entitlement, at that time, to an extension of time. The contractor-caused delay events that are alleged to have arisen in the period in question may also be impacted on the model and time analyzed. The resultant project end date will reflect the additional delay, if any, caused by the employer in that period. It will also demonstrate any concurrent periods of delay, meaning where both the employer and the contractor are both contributing to the same period of delay.

Following this initial analysis, the progress records are imposed on the planned model and the program analyzed again. The resulting end date now represents the overall delay to completion actually occurring in that window period. This will allow for excusable delay events, compensable delay events, culpable delay events, and contractor's mitigation due to changes in program or faster progress. From these two exercises (the planned model time-analyzed with delay events and the planned model analyzed with actual progress), the analyst is provided with information on causation which can then be debated as to dominance, concurrency, and mitigation [33]

Time impact analysis

The networked baseline schedule is first updated with progress to the point in time just before the delay event arose. If the updated schedule identifies a delay to completion this is registered. The delay event is then introduced into the schedule to establish the likely impact upon completion. The schedule is recalculated, and the new calculated delay to completion is registered. This is an iterative process repeated for each and every selected delay event.[11]

Time Impact Analysis (Windows Method)

The goal is to develop a snap shot picture of the project each time it experiences a major impact to the CPM schedule and accounts for the dynamic sequence of events and actual project history and compares impacts between the periodic snap shots. Fragnets (mini-schedules) are developed to indicate affects of the events not anticipated by the original schedule. A comprehensive time impact analysis will likely demonstrate the day for day increase in completion date, consumption of float, concurrency with another delay, recovery of time by acceleration or re-sequencing and accurately simulates project history. [84]

Isolated Delay Type

The IDT method [4] uses an as-planned schedule as a basis for comparison, and performs delay analysis based on the extracted analysis section schedule and considers the liabilities of contract parties explicitly. First, the IDT method divides the as-planned schedule into digestible periods. At the start of each analysis period, the IDT method modifies activity durations and relationships before and after the start point by inputting actual start dates, finish dates, and durations, and by maintaining original durations and logical relationships. This modified schedule is an adjusted as-planned schedule and the basis for delay comparisons.

The IDT method performs delay analysis for two independent viewpoints - of the owner and contractor. The IDT method inserts excusable and nonexcusable delays into the adjusted schedule to reflect the impact of delays from the contractor and owner perspectives, respectively. From one analysis viewpoint, delays caused by another party are imposed on the adjusted as-planned schedule. Therefore, the difference in time between project completion date on the adjusted as-planned schedule and the date after by inserting the delays is the duty to the opposition. By summarizing delay values at each analysis period, the IDT method can allocate liability to each contract party [103].

3.1.4.2 Advantages of TIA Group of Methods

A comprehensive time impact analysis will likely demonstrate the day for day increase in completion date, consumption of float, concurrency with another delay, recovery of time by acceleration or re-sequencing and accurately simulates project history [40].

Using the CPM algorithm, the time impact analysis method follows up on the project day-by-day from beginning to completion date, including consumption of float, concurrent delays, recovery time and acceleration [7].

3.1.4.3 Criticism of TIA Group Methods

1. TIA requires large amount of information.

   An as-planned schedule in CPM format is necessary; additionally, the schedule needs to be periodically updated. The projects that lack strict administrative procedures and/or updated schedules are not good candidates for this method [7].

2. TIA requires a lot of time.

   Time impact analysis consumes much more time compared to the other methods. Examining periodic updates is burdensome as actual data associated with many activities may need to be verified and compared for every updated period. Added or deleted activities have to be documented. In situations where time and budget are limited, time impact analysis may not be the method of preference [7].

3. TIA is fraud with manipulations.

   The analysis may be influenced by a variety of factors because time impact analysis is intricate as it determines accumulative results from a number of contemporaneous data [7]. Also in most variations, TIA is flexible with choice of time periods, and according to this decision results may be different. When a window period is treated separately from a delay event, the analysis may require approximation if the delay is divided between two window periods [35].

3.1.5 Other Delay Analysis Methods

3.1.5.1 Delay Analysis Method Using Delay Section (DAMUDS)

The delay analysis method using delay section (DAMUDS) [56] method has been proposed to overcome two limitations of existing windows methods; inadequate accounting of concurrent delay and inadequate accounting of time-shortened activities.

DAMUDS includes two new concepts, the delay section (DS) and the contractor's float (CF). The DS is a delay analysis time increment for dividing the delay-occurred duration into a single delay occurred period not overlapped and two or more delays- occurred periods overlapped [56]. CF represents the effort of a contractor to shorten the time of activities, thus reducing the total project duration [56].

The delay analysis procedures of DAMUDS are based on DS, CF and the analytical approach of traditional window analysis [51]. By identifying and calculating the delay impacts on project duration in each delay section, DAMUDS provides clear delay liabilities for the contractor and owner [56].

DAMUDS has a strong point in fixing time periods for analysis and recording float consumption. However same as contemporaneous delay analysis methods, it inaccurately portraits situation, when duration of delay events was known in advance and could have probably caused pacing delays. Acceleration is also not identified [51].

3.1.5.2 Daily Window Delay Analysis (DWDA)

Daily windows delay analysis (DWDA) proposed by Hegazy et al (2005) puts one day as a basic analysis unit. This approach considers the day-by-day fluctuation in critical path's along the project duration, and thus arrives at accurate and repeatable results for apportioning project delays among involved parties [42].

It has a strong point in possibility to address any change happened within a window. It can address consequental delays, accelerations or apportion delays to the lower, subcontractor level. And the fact, that duration of the period is fixed, reduces the possibility of manipulations. Results can be repeated.

DWDA is more accurate, detailed window technique, which overcomes drawbacks of contemporaneous period analysis and other conventional techniques. It still lacks ability to consider such situations as pacing delays, but among other methods DWDA is the closest conventional technique to the desirable delay analysis method.

3.2 Mathematical Formulation of Conventional Delay Analysis Methods

3.2.1 Basic Concepts of Delay Analysis

Delay Analysis begins with inspection and review of project records and related data in order to define and collect all necessary information required for analysis. Usually the availability and quality of this information becomes one of the critical factors in choosing the appropriate technique. For example, if the Project Baseline (or as-planned programme) is not detailed enough, or not realistic or in fact has never been created, methods like API cannot be applied, and there may be difficulties in applying CPA or TIA techniques.

3.2.1.1 The As-Planned Programme

The as-planned programme is supposed to show the Contractors true intentions, what activities were planned to be performed, when, in which sequence, etc. as detailed as possible. Usually it is made using available computer software based on the Critical Path Method.

As-planned programme Γ^0 includes a set of all planned activities $N = \{1, 2, \ldots, n\}$, for which the interrelations are know. In other words for each activity i , $i \in N$ there exist a set of preceding activities $\{-I\}$, which should start (or finish) before the start (or finish) of activity i and succeeding activities $\{+I\}$, which should start (or finish*) after the start (of finish) of activity i, $\{-I\}, \{+I\} \in N$.

* Depending on type of relationship between activities: Start-to-Start, Finish-to-Start, Finish-to-Finish, Start-to-Finish

Each activity $i \in N$ is characterized by planned start s_i and planned finish date f_i. We denote as $-i$ the closest predecessor to activity i If some activity j is in the set of preceding activities to activity i, i.e. $j \in \{-I\}$, and if it's finish date is the latest comparing to other activities in this set, then the activity j becomes the last activity which should be started (or finished) before activity i can be started (or finished). Activity j becomes a closest predecessor to activity i. In other words, the finish date of activity $-i$ is the maximum of finish dates of all activities in the set $\{-I\}$: $f_{-i} = max\{f_j; j \in \{-I\}\}$

Same way we define $+i$ as the closest successor of activity i, and the start date of activity $+i$ is equal to the minimum of start dates of all activities in the set $\{+I\}$: $s_{+i} = min\{s_j; j \in \{+I\}\}$

Planned Project Commencement date is denoted as S^0 and is equal to the start of the first activity in the Project: $S^0 = min\{s_i; i \in N\}$

Estimated (planned) Project completion date is denoted as E^0 and is equal to the finish date of the last activity in the Project: $E^0 = max\{f_i; i \in N\}$

Difference between planned Project commencement and completion dates defines the Planned Project Duration $\theta^0 = E^0 - S^0$;

Note: by default as planned start and finish we use the early dates.

3.2.1.2 The As-Built Programme

After collecting all as-planned information the Analyst should obtain the as-built data. It's availability, quality and specification becomes the second critical factor in choosing the appropriate DA technique. The best case scenario is having at hand an as-built programme used to control the project performance or reconstructed from daily or weekly reports.

The as-built programme is supposed to show how the Contractor actually executed the Project. What activities were performed, when and in which sequence. Usually the as-built programm is based on as-planned programme, but not necessarily.

As-built programme Γ^* includes a set of all actual activities $N^* = \{1, 2, \ldots, n^*\}$. For each activity i^* , $i^* \in N^*$ there exist a set of preceding activities $\{-I^*\}$, which have started (or finished) before the start (of finish) of activity i^* and succeeding activities $\{+I^*\}$, which have started (or finished) after the start (or finish) of activity i^*, $\{-I^*\}, \{+I^*\} \in N^*$.

Analogously to as-planned programme, the closest predecessor $-i^*$ and successor $+i^*$ of activity i^* are defined by the latest finish date of preceding activities and the earliest start date of succeeding activities: $f^*_{-i} = max\{f^*_j; j^* \in \{-I^*\}\}$ i^* if $s^*_{+i} = min\{s^*_j; j^* \in \{+I^*\}\}$

The actual start date of the first activity in the project becomes the Project Actual Commencement date and is denoted as S^*. $S^* = min\{s^*_i; i^* \in N^*\}$

The actual finish date of the last activity in the project becomes the Actual Project Completion Date and is denoted as E^*. $E^* = max\{f^*_i; i^* \in N^*\}$

Knowing the actual commencement and completion dates the Actual Project Duration can be calculated: $\theta^* = E^* - S^*$

3.2.1.3 The Adjusted As-planned programme

If the analyst has available as-planned and as-built programmes, just simple comparison of them itself can be considered as a delay analysis method, known as As-Planned versus As-Built. In case of small simple projects there might be no further calculations or analysis needed, however in regular or complicated situations such a comparison becomes only a part of a process of getting delay data and preparing for further steps in delay analysis.

Having two programmes in hand the analyst first should find if there is any correlation between them and to what extent. There must exist at least one identical activity in both programmes to link them together. In other words for some activity $i \in N$ there must be an equivalent activity $i^* \in N^*$, so that $i \equiv i^*$. However there is often a case when in as-built programme there are some actually performed activities which do not exist in as-planned programme (new, not planned activities): $j^* \in N^*$ while an equivalent activity $j \in \emptyset$ Or in as-planned programme there are some activities, which were dropped during project execution: $j \in N$ while $j^* \in \emptyset$

Because of the difference in sets of activities, using solely as-planned or as-built programme as a basis for analysis may lead to wrong results. If both programmes are available, the more trustful and increment analysis can be performed.

For such methods as TIA, CPA, DAMUDS the analyst would have to combine the as-planned ans as-built programmes and create a new basis for analysis - so called Modified or Adjusted As-Planned Programme. It reflects the original intention (plan) in most of the aspects (like project duration, activities dates, etc.) but at the same time is more prepared for analysis.

Adjusted As-Planned Programme Γ^0 includes all activities of original program and the as-built programme, i.e. $N^0 = N \cup N^*$. For each pair of activities $i \equiv i^*$ in the set of all activities in Adjusted Programme N^0 there is an identical activity i^0; If the set of planned activities is equivalent to the set of actual activities $N \equiv N^*$, then the set of activities in Modified programme will be equal to the set of planned activities $N^0 = N$, and each activity in modified programme will be equal to one in as-planned $i^0 = i$; However, as we mentioned before, it is a rare case when as-planned and as-built programmes have exactly same set of activities.

To create an Adjusted Programme the Analyst starts with the regular as-planned programme as the basis and then adds for each "new" actual activity j^* ($j^* \in N^*$, $j \in \emptyset$) the identical activity j^0 with zero duration $f_j^0 - s_j^0 = 0$. The start and finish dates of this activity is equal to the finish and start dates of its closest predecessor and successor: $s_j^0 = f_{-j}^0$ and $f_j^0 = s_{+j}^0$; The actual duration of this activity is considered as a delay event $\omega^k \in \Omega_j$ (will be discussed in the next section).

The Project Commencement and Estimated Completion dates of Adjusted Programme are equal to the original baseline or As-Planned Programme: $S^0 = S$ and $E^0 = E$, and hence the project duration in Γ^0 is equal to Γ: $\theta^0 = \theta$

3.2.1.4 Delay and Acceleration Information

After collecting all necessary as-planned and as-built information, the analyst focus on the delay issues. By comparison between Estimated and actual Project Completion Dates there can be drawn a conclusion whether the whole Project was delayed or accelerated.

Total Delay of the Project is equal to the positive difference between these dates: $D = (E^* - E^0)_+$; Total Acceleration of the Project is equal to negative: $A = (E^* - E^0)_-$;

Project can be either delayed or accelerated, but in both cases single delay events and acceleration periods may occur. In our interpretation of delay analysis methods the acceleration periods are considered as delay events with negative duration.

We denote the list of all delay events occurred (including accelerations) as Ω, $\Omega = \{\omega^1, \omega^2, \ldots, \omega^k, \ldots, \omega^K, \}$, where ω^k is a single delay event.

The subset of delay events, which had an impact on some activity i, we denote as $\Omega_i \in \Omega$. Each delay event ω^k impacts only one activity, i.e. if $\omega^k \in \Omega_i$, then duration of this event on any other activity $j \in \{N \setminus i\}$ is equal zero: $e^k_j - b^k_j = 0$.

Note, than even if in the project there was some delay event which affected multiple activities (for example, adverse weather conditions), in fact the real impact of this delay event on each activity may be different. For the accuracy of analysis we create separate delay events for each affected activity.

Delay event ω^k on activity i is characterized by beginning b^k_i and ending e^k_i dates. Accelerations are characterized by $b^k_i < 0$ and $e^k_i < 0$. If the Project has some dropped activity, which was planned, but not executed (i.e. $j \in N$ while $j^* \in \emptyset$), then in some DA methods it is considered as acceleration period and is denoted as delay event ω^k_j with negative duration: $e^k_j - b^k_j < 0$.

In Delay Analysis the order in which delay events are analyzed plays a critical role. Naturally the closer the analysis is to how events actually happened the more accurate result it gives. All delay events in Ω are put in chronological order: $\mid b^k_i \mid < \mid b^{k+1}_i \mid$;

Even though single delay event causes delay to some activity it does not necessarily delays the whole Project. This type of delay is called noncritical, however for the sake of accuracy delay events should be analyzed regardless to their criticality.

For each delay event (critical or not) there should be defined a responsible party. So in set of all delay events Ω there exist several subsets of delay events related to particular parties. For example, all delay events caused by the Owner form the subset $\Omega_O \in \Omega$.

3.2.1.5 Time Periods or 'Windows'

Delay events can be analyzed considering its impact at one time or by time periods. In latter approach overall Project duration is divided into several time periods. The length of each period may depend on delay events nature, analyst's subjective decision or the type of DA Method. For example in Daily Analysis all delay events are analyzed day by day. The project execution is reconstructed day by day reflecting how things have actually happen. With all necessary data available this method might be considered being the most precise and accurate, however time and cost it requires makes it not efficient, especially if same results could be obtained by using other methods.

Set of time periods $\Theta = \{1, 2, \ldots, \tau, \ldots, T\}$.

Each period is characterized by open o^τ and close dates c^τ, duration of particular time period becomes $t^\tau = c^\tau - o^\tau$

Delay events Ω are cut by these time periods, forming sets of parts of delay events. $\Omega^\tau = \{\omega^{1,\tau}, \omega^{2,\tau}, \ldots, \omega^{k,\tau}, \ldots, \omega^{K,\tau},\}$ is a set of all parts of delay events occurred in a time period τ.

Delay events considered within the time periods form the following Matrix (M1):

Ω	Ω^1	Ω^2	$\cdots$	Ω^τ	$\cdots$	Ω^T
ω^1	$\omega^{1,1}$	$\omega^{1,2}$	$\cdots$	$\omega^{1,\tau}$	$\cdots$	$\omega^{1,T}$
ω^2	$\omega^{2,1}$	$\omega^{2,2}$	$\cdots$	$\omega^{2,\tau}$	$\cdots$	$\omega^{2,T}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
ω^k	$\omega^{k,1}$	$\omega^{k,2}$	$\cdots$	$\omega^{k,\tau}$	$\cdots$	$\omega^{k,T}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
ω^K	$\omega^{K,1}$	$\omega^{K,2}$	$\cdots$	$\omega^{K,\tau}$	$\cdots$	$\omega^{K,T}$

Where $\omega^{k,\tau}$ is the part of delay event ω^k in time period τ. And Ω can be defined as set of all delay events happen in the Project $\Omega = \{\omega^1, \omega^2, \ldots, \omega^k, \ldots, \omega^K\}$ or as a set of subsets of all parts of delay events occurred within a particular period τ, i.e. $\Omega =$

$\{\Omega^1, \Omega^2, \ldots, \Omega^\tau, \ldots, \Omega^T\}$, where $\Omega^\tau = \{\omega^{1,\tau}, \omega^{2,\tau}, \ldots, \omega^{k,\tau}, \ldots, \omega^{K,\tau}\}$

Briefly

$$\Omega = \sum_{k=1}^{K} \sum_{\tau=1}^{T} \omega^{k,\tau}$$

Delay event $\omega^k \in \Omega^\tau$ if there exist an intersection $[b_i^k; e_i^k] \cap [o^\tau; c^\tau]$;

Each part of delay event $\omega^{k,\tau}$ is characterized by beginning $b_i^{k,\tau}$ and ending $e_i^{k,\tau}$ dates;

If some delay event ω^k on activity i has occurred in a time period τ, then $\omega_i^{k,\tau}$ has a nonzero duration (i.e. $e_i^{k,\tau} - b_i^{k,\tau} \neq 0$).

$b_i^{k,\tau}$ is equal to the beginning of delay event ω^k , i.e. $b_i^{k,\tau} = b_i^k$ if it started within the period τ, or to the open date of period τ $b_i^{k,\tau} = o^\tau$, if it started in one of the previous periods.

$b_i^{k,\tau} = max\{b_i^k; o^\tau\}$, in other words:

$$b_i^{k,\tau} = \begin{cases} b_i^k & \text{if } b_i^k \in [o^\tau; c^\tau) \\ o^\tau & \text{otherwise} \end{cases}$$

$e_i^{k,\tau}$ is equal to the ending of delay event ω^k , i.e. $e_i^{k,\tau} = e_i^k$ if it finished within the period τ, or to the end date of period τ: $e_i^{k,\tau} = c^\tau$, if it finishes in one of the next periods.

$e_i^{k,\tau} = min\{e_i^k; c^\tau\}$, in other words:

$$e_i^{k,\tau} = \begin{cases} e_i^k & \text{if } e_i^k \in (o^\tau; c^\tau] \\ c^\tau & \text{otherwise} \end{cases}$$

If $\omega^k \notin \Omega^\tau$, then beginning $b_i^{k,\tau} = 0$ and ending date $e_i^{k,\tau} = 0$;

3.2.1.6 Certain and uncertain delay events

For some methods, like Time Impact Analysis (TIA), Delay events Ω may be divided into two groups:

- *certain delay events* Ω_K , for which its ending dates (and hence the impact on the Project finish date) were predicted or known during the Project execution.

- *uncertain delay events* Ω_U , for which the ending dates could not be predicted during the Project execution and were known after (post factum).

Certain delay events may have influence on the choice of time periods. For example, in TIA the open o^τ and close dates c^τ of time periods could be chosen depending on the beginning and ending dates of the certain delay events Ω_K.

For the first period: $o^1 = S^0$, $c^1 = b_i^m$, where $\omega^m \in \Omega_K$;

For the second period: $o^2 = c^1$, $c^2 = b_i^{m+}$, where $\omega^{m+} \in \Omega_K$ and is chronologically next after the event ω^m;

For the time period τ: $o^\tau = c^{\tau-1}$, $c^\tau = b_i^k$, where $\omega^k \in \Omega_K$ and is chronologically next after the event ω^{k-}, which was considered in time period $\tau - 1$;

For the time period T: $o^T = c^{T-1}$, $c^T = E^*$.

Certain delay event $\omega^k \in \Omega_K^\tau$ if the open date of time period τ is equal to the beginning of delay event ω^k on activity i, i.e. if $o^\tau = b_i^k$. Certain delay event can belong only to one time period. It is denoted as $\omega^{k,\tau}$ and characterized by beginning $b_i^{k,\tau} = b_i^k$ and ending dates $e_i^{k,\tau} = e_i^k$.

If $\omega^k \notin \Omega^\tau$, then duration of $\omega^{k,\tau}$ is equal zero. And hence $b_i^{k,\tau} = 0$ and $e_i^{k,\tau} = 0$.

After the open and close dates of each time period are calculated, the uncertain delay events are "cut" into parts so that for each delay event $\omega^u \in \Omega_U$ in each time period τ there is a part of delay event $\omega^{u,\tau}$.

Uncertain delay event $\omega^u \in \Omega_U^\tau$ if there exist an intersection $[b_i^u; e_i^u] \cap [o^\tau; c^\tau]$;

1. If $\omega^u \in \Omega_U^\tau$, then $e^{u,\tau} - b^{u,\tau} \neq 0$

   Each nonzero part of uncertain delay event $\omega^{u,\tau} \neq 0$ is characterized by beginning $b_i^{u,\tau}$ and ending $e_i^{u,\tau}$ dates;

   $b_i^{u,\tau} = max\{b_i^u; o^\tau\}$, in other words:

$$b_i^{u,\tau} = \begin{cases} b_i^u & \text{if } b_i^u \in [o^\tau; c^\tau) \\ o^\tau & \text{otherwise} \end{cases}$$

   $b_i^{u,\tau}$ is equal to the beginning of delay event ω^u , i.e. $b_i^{u,\tau} = b_i^u$ if it started within the period τ, or to the open date of period τ $b_i^{u,\tau} = o^\tau$, if it started in one of the previous periods.

$e_i^{u,\tau} = min\{e_i^u; c^\tau\}$, in other words:

$$e_i^{u,\tau} = \begin{cases} e_i^u & \text{if } e_i^u \in (o^\tau; c^\tau] \\ c^\tau & \text{otherwise} \end{cases}$$

$e_i^{u,\tau}$ is equal to the ending of delay event ω^k , i.e. $e_i^{u,\tau} = e_i^u$ if it finished within the period τ, or to the end date of period τ: $e_i^{u,\tau} = c^\tau$, if it finishes in one of the next periods.

2. If $\omega^u \notin \Omega^\tau$, then duration of $\omega^{u,\tau}$ is equal zero,and hence beginning $b_i^{u,\tau} = 0$ and ending date $e_i^{u,\tau} = 0$.

3.2.2 As-Planned Impacted (API)

Brief description:

Delay event(s) are added into the As-Planned program (or Baseline) in chronological order and then the project completion date is reanalyzed until all of the delays have been impacted.

Prerequisites:

- The As-planned programm Γ^0
- Some of the As-built data, such as Actual Project start S^* and finish dates F^* and Actual Project Duration $\theta^* = E^* - S^*$
- Total Delay of the Project D, $D = (E^* - E^0)_+$
- List of delay events $\Omega = \{\omega^1, \omega^2, \ldots, \omega^k, \ldots, \omega^K\}$

Algorithm:

The as-planned programm Γ^0 is periodically updated by adding delay events Ω to the schedule in chronological order. After each ω^k, is added, all activities dates are recalculated according to formulas (3.1) and (3.2) to make a new impacted programm Γ^k.

$$s_i^k = s_i^{k-1} + (f_{-i}^k - s_i^{k-1})_+ \tag{3.1}$$

s_i^k - new start date of activity i after the impact of delay event ω^k;

s_i^{k-1} - start date of activity i after the impact of previous delay event ω^{k-1};

if $k = 1$, then $s_i^{k-1} = s_i^0$;

f_{-i}^k - finish date of activity i after the impact of delay event ω^k;

$(f_{-i}^k - s_i^{k-1})_+$ - delay to activity i caused by its closest predecessor $-i$, which was already impacted by ω^k;

$$f_i^k = s_i^k + (e_i^k - b_i^k) + (f_i^{k-1} - s_i^{k-1,}) \tag{3.2}$$

f_i^k - new finish date of activity i after the impact of delay event ω^k;

$(e_i^k - b_i^k)$ - duration of delay event ω^{k-1};

$(f_i^{k-1} - s_i^{k-1})$ - duration of activity i after the impact of the previous delay event ω^{k-1}, but before the impact of delay event ω^k; In other words duration of activity i in the programm Γ^{k-1};

if $k = 1$, then $s_i^{k-1} = s_i^0$ and $f_i^{k-1} = f_i^0$, where s_i^0 and f_i^0 - are the original start and finish dates of activity i in programm Γ^0;

After the impacted programm Γ^k is formed, the Delay to the Project, caused by delay event ω^k, can be calculated by formula (3.3):

$$D^k = (E^k - E^{k-1}) \tag{3.3}$$

$E^k = max\{f_i^k; i \in N\}$ - Estimated Project completion date of the programm Γ^k;

$E^{k-1} = max\{f_i^{k-1}; i \in N\}$ - Estimated Project completion date of the programm Γ^{k-1};

if $k = 1$, then $f_i^{k-1} = f_i^0$ and hence $E^{k-1} = E^0$;

Analyzing Results: Results from each recalculation form the following table:

Ω	ω^1	ω^2	$\cdots$	ω^k	$\cdots$	ω^K
D	D^1	D^2	$\cdots$	D^k	$\cdots$	D^K

Where each $D^k \neq 0$ demonstrates the responsibility of particular delay event ω^k;

According to origin of ω^k the responsible party for delay D^k can be determined.

3.2.3 As-Planned But-For (APBF)

Brief description:

This method is a modification of the API. It follows same philosophy, but only DE related to one party are considered.

We use the interpretation of this method given by Tony Farrow:[1]

"The next method is the as-planned but for method and in this case, the analyst impacts the planned baseline program with the assessed implications of the events a party considers it is responsible for and the combined influences of these are analyzed. The impacted completion date is then compared with the as-built completion date (that is, when the project was actually completed) and the difference is said to be how much earlier the project could have finished but for all the other delay events (imposed by the other party) but which have not been analyzed".

Prerequisites:

- The As-planned programm Γ^0
- Some of the As-built data, such as Actual Project start S^* and finish dates F^* and Actual Project Duration $\theta^* = E^* - S^*$
- Total Delay of the Project D, $D = (E^* - E^0)_+$
- List of all delay events, related to one party, from which perspective the analysis is performed. For example, Contractor's delay events $\Omega_C = \{\omega_c^1, \omega_c^2, \ldots, \omega_c^k, \ldots, \omega_c^K\}$;

[1] Tony Farrow: Developments in the Analysis of Extensions of Time, ASCE, July 2007.

Algorithm:

The as-planned programm Γ^0 is periodically updated by adding delay events Ω_C to the schedule in chronological order. After each ω_c^k, is added, all activities dates are recalculated according to formulas (3.1) and (3.2) to make a new impacted programm Γ_c^k.

After the impacted programm Γ_c^k is formed, the Delay to the Project, caused by delay event ω_c^k, can be calculated by formula (3.3).

Analyzing Results:

Results from each recalculation form the following table:

Ω_C	ω_c^1	ω_c^2	$\cdots$	ω_c^k	$\cdots$	ω_c^K
D_C	D_c^1	D_c^2	$\cdots$	D_c^k	$\cdots$	D_c^K

Where each $D_c^k \neq 0$ demonstrates the responsibility of particular delay event ω_c^k;

D_C represents the responsibility of the analyzed party (Contractor).

$$D_C = \sum_{k=1}^{K} D_c^k$$

The difference between D_C and D (or between E_c^K and E^*) is meant to show how much earlier the Project could have finished but for all other events related to other parties which were not analyzed.

3.2.4 Collapsed As-built (CollAB)

Brief description:

This technique is the reverse of The As-Planned impacted. Delay Events are subtracted from the As-built program. The difference in overall program duration before and after this subtraction is said to represent the period of critical delay by the particular delay event removed.

Prerequisites:

- The As-built programm Γ^0.
- Planned Project start S and completion dates E, Planned Project duration $\theta = E - S$;

- Total Delay of the Project D, $D = (E^* - E^0)_+$
- List of delay events $\Omega = \{\omega^1, \omega^2, \ldots, \omega^k, \ldots, \omega^K\}$
- Delay sections:

Algorithm:

The as-built programm Γ^0 is periodically updated by subtracting delay events Ω from the schedule in *reverse* chronological order. After each ω^k, is subtracted, all activities dates are recalculated according to formulas (3.4) and (3.5) to make a new collapsed programm Γ^k.

$$s_i^k = s_i^{k+1} - (s_i^{k+1} - f_{-i}^k) \tag{3.4}$$

s_i^k - new start date of activity i after the subtraction of delay event ω^k;

s_i^{k+1} - start date of activity i after the subtraction of previously analyzed delay event ω^{k+1};

if $k = K$, then $s_i^{k+1} = s_i^0$;

f_{-i}^k - finish date of activity $-i$ after the subtraction of delay event ω^k;

$(s_i^{k+1} - f_{-i}^k)$ - change in the start date of activity i caused by its closest predecessor $-i$, which dates were already changed by subtraction of ω^k;

$$f_i^k = s_i^k - (e_i^k - b_i^k) + (f_i^{k+1} - s_i^{k+1,}) \tag{3.5}$$

f_i^k - new finish date of activity i after the subtraction of delay event ω^k;

$(e_i^k - b_i^k)$ - duration of delay event ω^k;

$(f_i^{k+1} - s_i^{k+1})$ - duration of activity i after the subtraction of the previously analyzed delay event ω^{k+1}, but before the subtraction of delay event ω^k; In other words duration of activity i in the programm Γ^{k+1};

if $k = K$, then $s_i^{k+1} = s_i^0$ and $f_i^{k+1} = f_i^0$, where s_i^0 and f_i^0 - are the actual start and finish dates of activity i in programm Γ^0;

After the collapsed programm Γ^k is formed, the Delay to the Project, caused by delay event ω^k, can be calculated by formula (3.6):

$$D^k = (E^{k+1} - E^k) \tag{3.6}$$

$E^{k+1} = max\{f_i^{k+1}; i \in N^0\}$ - Estimated Project completion date of the programm Γ_{CollAB}^{k+1};

$E^k = max\{f_i^k; i \in N^0\}$ - Estimated Project completion date of the programm Γ^k;

if $k = K$, then $f_i^{k+1} = f_i^0$ and hence $E^{k+1} = E^0$;

Analyzing Results:

Results from each recalculation form the following table:

Ω	ω^1	ω^2	$\cdots$	ω^k	$\cdots$	ω^K
D	D^1	D^2	$\cdots$	D^k	$\cdots$	D^K

Where each $D^k \neq 0$ demonstrates the responsibility of particular delay event ω^k;

According to origin of ω^k the responsible party for delay D^k can be determined.

3.2.5 As-built But-For (ABBF)

Brief description:

Delay event(s) related to one party are added into the As-Planned program (or Baseline) in chronological order and then the project completion date is reanalyzed until all of the delays of this party have been impacted.

Prerequisites:

- The As-built programm Γ^0.
- Planned Project start S and completion dates E, Planned Project duration $\theta = E - S$;

- Total Delay of the Project D, $D = (E^* - E^0)_+$
- List of all delay events, related to one party, from which perspective the analysis is performed. For example, Contractor's delay events $\Omega_C = \{\omega_c^1, \omega_c^2, \ldots, \omega_c^k, \ldots, \omega_c^K, \}$;
- Delay sections:

Algorithm:

The as-planned programm Γ^0 is periodically updated by adding delay events Ω_C to the schedule in chronological order. After each ω_c^k, is added, all activities dates are recalculated according to formulas (3.4) and (3.4) to make a new impacted programm Γ_c^k.

After the impacted programm Γ_c^k is formed, the Delay to the Project, caused by delay event ω_c^k, can be calculated by formula (3.6):

Analyzing Results:

Results from each recalculation form the following table:

Ω_C	ω_c^1	ω_c^2	$\cdots$	ω_c^k	$\cdots$	ω_c^K
D_C	D_c^1	D_c^2	$\cdots$	D_c^k	$\cdots$	D_c^K

Where each $D_c^k \neq 0$ demonstrates the responsibility of particular delay event ω_c^k;

D_C represents the responsibility of the analyzed party (Contractor).

$$D_C = \sum_{k=1}^{K} D_c^k$$

The difference between D_C and D (or between E_c^K and E^*) is meant to show how much earlier the Project could have finished but for all other events related to other parties which were not analyzed.

3.2.6 Contemporaneous Period Analysis (CPA)

Brief description:

The as-planned program in overall project duration is divided into periods. Then the analyst updates this program within the first period using information about all delays and accelerations, which happened in that period. The remaining as-planned schedule is maintained beyond this period. The delay or acceleration arose in current period is represented by the difference between the project completion date after updating this period and the project completion date from the previous period.

Prerequisites:

- The Modified As-planned programm Γ^0, based on as-planned and as-built programms.
- Total Delay of the Project D, $D = (E^* - E^0)_+$
- List of delay events $\Omega = \{\omega^1, \omega^2, \dots, \omega^k, \dots, \omega^K, \}$
- Set of time periods $\Theta = \{1, 2, \dots, \tau, \dots, T\}$
- the matrix of delay events considered within the time periods (M1).

Algorithm:

The original modified as-planned programm Γ^0 is periodically updated by adding parts of delay events $\omega^{k,\tau}$ to the schedule in chronological order. After each $\omega^{k,\tau}$ is added, all activities dates are recalculated according to formulas (3.7) and (3.8) to make a new impacted programm $\Gamma^{k,\tau}$.

$$s_i^{k,\tau} = s_i^{k-1,\tau} + (f_{-i}^{k,\tau} - s_i^{k-1,\tau})_+ \tag{3.7}$$

$s_i^{k,\tau}$ - new start date of activity i after the impact of delay event $\omega^{k,\tau}$;

$s_i^{k-1,\tau}$ - start date of activity i after the impact of previous delay event $\omega^{k-1,\tau}$;

if $k = 1$, then $s_i^{k-1,\tau} = s_i^{K,\tau-1}$, where $s^{K,\tau-1}$ - the last delay event in the previous period $\tau - 1$;

if $\tau = 1$, then $s_i^{K,\tau-1} = s_i^0$, where s_i^0 - the original start date of activity i in programm Γ^0_{CPA};

$f_{-i}^{k,\tau}$ - finish date of activity $-i$ after the impact of delay event $\omega^{k,\tau}$;

$(f_{-i}^{k,\tau} - s_i^{k-1,\tau})_+$ delay to activity i caused by its closest predecessor $-i$, which was already impacted by $\omega^{k,\tau}$;

$$f_i^{k,\tau} = s_i^{k,\tau} + (e_i^{k,\tau} - b_i^{k,\tau}) + (f_i^{k-1,\tau} - s_i^{k-1,\tau}) \tag{3.8}$$

$f_i^{k,\tau}$ - new finish date of activity i after the impact of delay event $\omega^{k,\tau}$;

$(e_i^{k,\tau} - b_i^{k,\tau})$ - duration of delay event $\omega^{k-1,\tau}$; If $(e_i^{k,\tau} - b_i^{k,\tau}) < 0$, then acceleration took place;

$(f_i^{k-1,\tau} - s_i^{k-1,\tau})$ - duration of activity i after the impact of the previous delay event $\omega^{k-1,\tau}$, but before the impact of delay event $\omega^{k,\tau}$; In other words duration of activity i in the programm $\Gamma^{k-1,\tau}$;

if $k = 1$, then $s_i^{k-1,\tau} = s_i^{K,\tau-1}$ and $f_i^{k-1,\tau} = f_i^{K,\tau-1}$, where $\omega^{K,\tau-1}$ - the last delay event in the previous period $\tau - 1$;

if $\tau = 1$, then $s_i^{K,\tau-1} = s_i^0$ and $f_i^{K,\tau-1} = f_i^0$, where s_i^0 and f_i^0 - are the original start and finish dates of activity i in programm Γ^0;

After the impacted programm $\Gamma^{k,\tau}$ is formed, the Delay (or Acceleration) to the Project, caused by delay event $\omega^{k,\tau}$, can be calculated by formula 3.9:

$$D^{k,\tau} = (E^{k,\tau} - E^{k-1,\tau}) \tag{3.9}$$

Where $E^{k,\tau} = max\{f_i^{k,\tau}; i \in N\}$ - Estimated Project completion date of the programm $\Gamma^{k,\tau}$;

$E^{k-1,\tau} = max\{f_i^{k-1,\tau}; i \in N\}$ - Estimated Project completion date of the programm $\Gamma^{k-1,\tau}$;

if $k = 1$, then $f_i^{k-1,\tau} = f_i^{K,\tau-1}$ and hence $E^{k-1,\tau} = E^{K,\tau-1}$;

if $\tau = 1$, then $f_i^{K,\tau-1} = f_i^0$ and hence $E^{K,\tau-1} = E^0$;

Analyzing Results:

Results from each recalculation form the following matrix (M2):

D	D^1	D^2	$\cdots$	D^τ	$\cdots$	D^T
d^1	$D^{1,1}$	$D^{1,2}$	$\cdots$	$D^{1,\tau}$	$\cdots$	$D^{1,T}$
d^2	$D^{2,1}$	$D^{2,2}$	$\cdots$	$D^{2,\tau}$	$\cdots$	$D^{2,T}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
d^k	$D^{k,1}$	$D^{k,2}$	$\cdots$	$D^{k,\tau}$	$\cdots$	$D^{k,T}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
d^K	$D^{K,1}$	$D^{K,2}$	$\cdots$	$D^{K,\tau}$	$\cdots$	$D^{K,T}$

Where each $D^{k,\tau} \neq 0$ demonstrates the responsibility of particular delay event ω^k within a period τ;

$D^\tau = \{D^{1,\tau}, D^{2,\tau}, \ldots, D^{k,\tau}, \ldots, D^{K,\tau}, \}$ - Project Delay occurred in a period τ;

$d^k = \{D^{k,1}, D^{k,2}, \ldots, D^{k,\tau}, \ldots, D^{k,T}, \}$ - delay to the Project caused by delay event ω^k;

According to origin of ω^k the responsible party for delay D^k can be determined.

3.2.7 Time Impact Analysis (TIA)

Brief description:

This method is interpreted as a combination of the CPA and the API methods. The as-planned programm is updated periodically by adding *certain* delay events to the schedule (like in API) and adding delay periods or parts of *uncertain* delay events, truncated by time periods (like in CPA).

The delay or acceleration arose in current period is represented by the difference between the project completion date after updating this period and the project completion date from the previous period.

For this method two types of delay events should be considered.

During the project execution when the delay occurs in some cases it may be possible to predict delay period ending date. So when at least the approximate impact on the Project is known, the parties take actions having this information in hand. For example, if one of two parallel activities is delayed for known period, the other activity becomes noncritical with some float, and parties can make plans on this float and use it. We call this type of delay events *certain.*

So when the analyst goes back in time and tries to follow the events as close as possible to how things originally happened, it becomes important to keep in mind whether the impact of delay event was known or predicted during the project execution or not.

Delay events, which could not be predicted, we call *uncertain.*

Prerequisites:

- The Adjusted As-planned programm Γ^0, based on as-planned and as-built programms.
- Total Delay of the Project D, $D = (E^* - E^0)_+$
- List of delay events $\Omega = \{\omega^1, \omega^2, \ldots, \omega^k, \ldots, \omega^K\}$
- List of certain Ω_K, $\Omega_K \in \Omega$ and uncertain delay events Ω_U, $\Omega_U \in \Omega$;
- Set of time periods $\Theta = \{1, 2, \ldots, \tau, \ldots, T\}$, chosen depending on certain delay events.
- the matrix of delay events considered within the time periods (M1).

Algorithm:

The modified as-planned programm Γ^0 is periodically updated by adding delay events $\omega^{k,\tau}$ to the schedule in chronological order. After each $\omega^{k,\tau}$ is added, all activities dates are recalculated using formulas (3.7) and (3.8) to make a new impacted programm $\Gamma^{k,\tau}$.

After the impacted programm $\Gamma^{k,\tau}$ is formed, the Delay (or Acceleration) to the Project, caused by delay event $\omega^{k,\tau}$, can be calculated by formula (3.9).

Analyzing Results:

Results from each recalculation form the matrix (M2), where each $D^{k,\tau} \neq 0$ demonstrates the responsibility of particular delay event ω^k within a period τ; According to origin of ω^k the responsible party for delay D^k can be determined.

Chapter 4

DESIRABLE METHOD OF DELAY ANALYSIS

This chapter investigates the key elements of delay analysis, which form its methodological framework and algorithms. Desirable properties of delay events and delay analysis options are established and discussed.

4.1 Extended Concept of Delay Event

4.1.1 Delay Event is attributed to a single activity

Term 'Delay event' in practice is often used to mark some happening, occurrences, literally 'event', that had delayed the whole Project or its part. However in Delay Analysis the term should be understood as a **period of delay on a single activity attributed to particular party.**

One happening or risk event, occurred in the Project may have impact on several activities. The effect may be concurrent, i.e. fall in the same period of time, or may have influence at different moments. For example, severe weather conditions may suspend all works on the site for a certain period of time. But change in government authority regulations may affect documentation approval activities through the whole project.

Figure 4.1 shows the situation when two parallel non-sequential activities are impacted by the same delay event (weather conditions), but because of the difference in activities starting and ending dates, the actual period of delay is also different.

Let us denote the list of all delay events occurred in the Project as Ω, $\Omega = \{\omega^1, \omega^2, \ldots, \omega^k, \ldots, \omega^K, \}$,

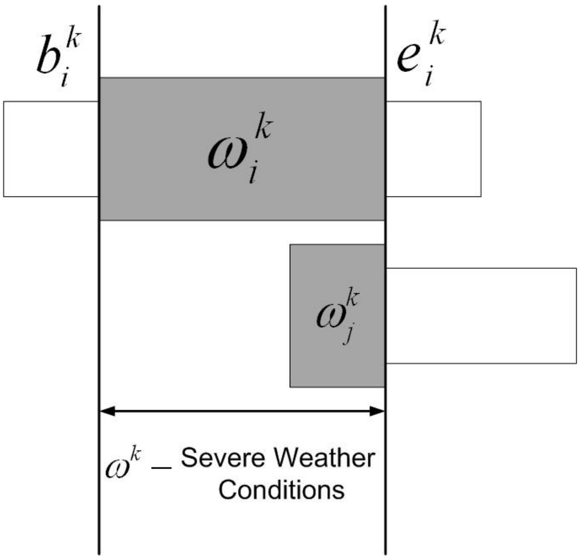

Figure 4.1: Impact of Delay Event on two activities

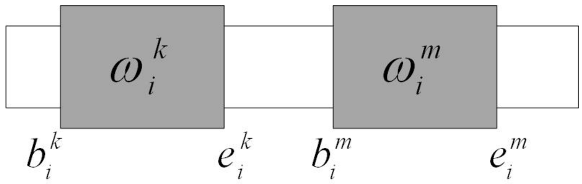

Figure 4.2: Activity impacted by two delay events

where ω^k is a single delay event.

By definition each delay event impacts at least one activity and can be characterized by beginning & ending dates and duration. The subset of delay events, which had an impact on some activity i is denoted as $\Omega_i \in \Omega$. If delay event ω^k had an impact on activity i, i.e. if $\omega^k \in \Omega_i$, then its duration can be defined as $\omega_i^k = e_i^k - b_i^k$.

In delay situation, illustrated by Figure 4.1, $\omega_i^k \neq \omega_j^k$. If two activities, influenced by the same delay event, are performed in separated parts of the project and delay periods on these activities do not coincide, i.e. when $b_i^k \neq b_j^k$ and $e_i^k \neq e_j^k$, it is naturally better to consider it as two independent delay events ω^k and ω^m with durations . $\omega_i^k = e_i^k - b_i^k$ and $\omega_j^m = e_j^m - b_j^m$.

To make an analysis more transparent it is suggested and further assumed that **each delay event is related only to one activity.**In other words if $\omega^k \in \Omega_i$, then duration of this event on any other activity $j \in \{N \setminus i\}$ is equal zero: $e_j^k - b_j^k = 0$

4.1.2 On a single activity there can be several Delay Events

As was established in the previous paragraph, each delay event should be assigned to only one activity, however one activity can be impacted by several delay events (Fig. 4.2).
$\{\omega^k, \omega^m\} \in \Omega_i$, where Ω_i is the set of all delay events, which had an impact on activity i, $\Omega_i \in \Omega$.

Delay events on a single activity should be clearly identified in terms of beginning and

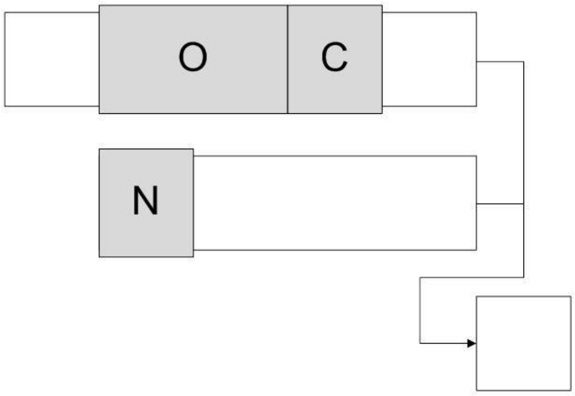

Figure 4.3: The As-Built Schedule with delay events (Owner's, Contractor's and Neutral)

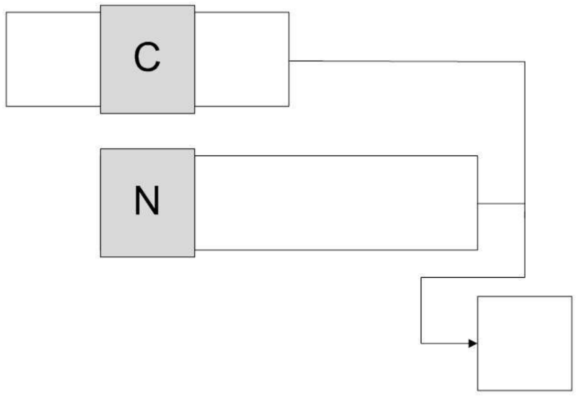

Figure 4.4: Subtraction of Owner caused Delay Event

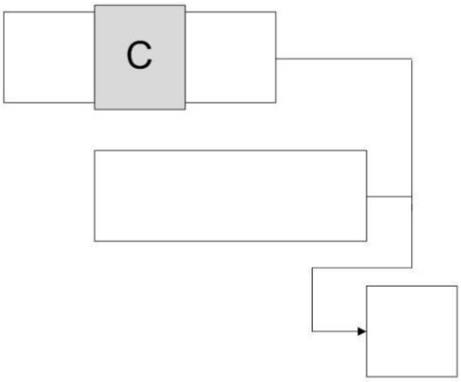

Figure 4.5: Subtraction of all excusable delay events

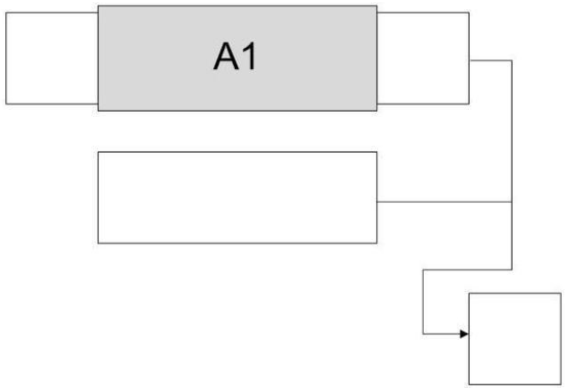

Figure 4.6: Project delay as a fault of delayed activity

ending dates so that there is no intersection between events. $(b_i^k; e_i^k) \cap (b_i^m; e_i^m) = \emptyset$

4.1.3 Project delay is an effect of delay event

There are three ways to assess delay to the Project Completion:

(a) As a result of particular delay event(s).

Delay Events one by one or in a 'single shot' are added to as-planned (or subtracted from as-built programme), and the difference between the resulting entitlement schedule and the schedule chosen as a baseline for analysis (e.g. As-Planned Programme for API analysis) represents the portion of delay to completion caused by these delay events. This approach is used in "non-window" conventional methods.

Figure 4.3 illustrates an as-built programme with three delay events. Figures 4.4 demonstrate the result of subtraction of Owner caused delay event and Figure 4.5 result of subtraction of all excusable delays in 'one shot'.

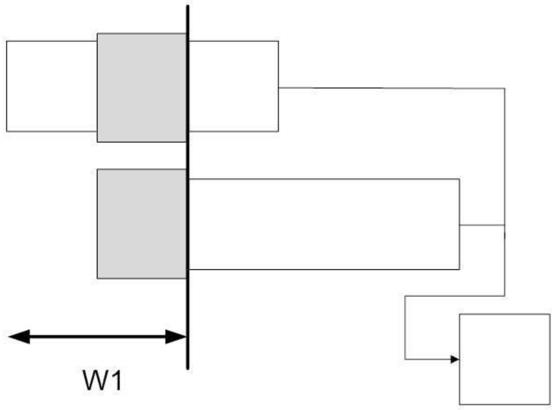

Figure 4.7: Window Analysis

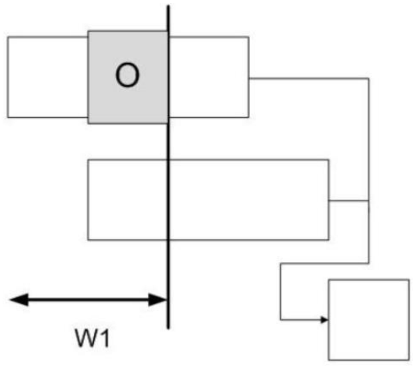

Figure 4.8: Owner-caused delay event 'cut' by Window

(b) As a fault of single delayed activity.

The responsibility for the Project delay is distributed among activities, and then the party in charge of this activity is considered liable for the relevant portion of project delay. This approach is used in the Project Game Concept [29], however the main limitation of this assumption is inability to deal with situations when one activity is affected by several delay events, caused by different parties (as illustrated in Fig. 4.6)

(c) As a consequence of multiple delays occurred within a particular period of time or window.

The Project schedule is updated on a period based, and the change in project completion date before and after this update represents the amount of delay occurred within a period in focus (As shown in Fig. 4.7). The strongest side of this approach is a possibility of movement from the beginning to the end of the project modeling the situations, arisen at certain time frames or windows. It is used in all window analysis type of methods (further about these methods in Chapter 4). However distribution of responsibility within a window may become a difficult tusk.

The most desirable solution lies in combination of the first and third approaches. Delay events should be considered independently one by one, but within certain time periods. If duration of some delay event goes beyond the period in focus, it may be 'cut' in parts and considered separately (As shown in Fig. 4.8). This approach is applied, for example, in DAMUDS [56].

4.1.4 Certain and Uncertain Delay events

Naturally the better delay analysis represents the real project situation at the moment of delay, the more reliable its results are. One of the weaknesses of 'window' approach, which divides the Project duration into several time periods, is the probability to contradict with the real situation. Though the analysis is usually performed chronologically, attempting to reconstruct the actual project flow, it consist an assumption, that the period of particular delay at that moment was known only till the end of the analyzed window.

In the Figure 4.8 the duration of the Owner-caused delay event in the first step is analyzed only till the end of first window, and the conclusion is that it did not cause any delay to the Project completion, just consumed an available float. At the same time an impact approach (described in the previous paragraph) has a contrary assumption.

A desirable delay analysis method should be able to consider both situations. It is suggested in the present Study, that prior to analysis all delay events should be catheterized and labeled in terms of their certainty. So that there will be two types of delay events:

- *certain delay events* , for which its ending dates (and hence the impact on the Project finish date) were predicted or known during the Project execution.
- *uncertain delay events* , for which the ending dates could not be predicted during the Project execution and were known after (post factum).

Notification of delay, the moment when the fact of current of future delay was revealed, also should be identified and registered. If necessary, delay period can be broken into two delay events: uncertain (before the fact of delay was revealed) and certain (when the delay ending dates were predicted).

4.1.5 Delay event with fixed and 'floating' dates

One of the weaknesses of conventional methods, where delay events are subtracted from the as-built schedule (or added to as-planned), is the assumption, that other delay events would have happened anyway. Zack (2001) puts the question "would the work of the project have been done in the same manner had the delays not occurred?" as one of the important challenges to But-for Schedules Analysis. [107]

Figures 4.3 and 4.4 illustrate the analysis of the owner-caused delay event under assumption, that if it did not occure, the contractor-caused delay event still would have impacted this activity. Note, that the dates of this Contractor-caused delay event have changed, which often can be seen in conventional methods application examples.

Same as with the certainty and uncertainty, delay events according to their 'timing' nature can be categorized into two groups:

- *Delay events with fixed dates*, which has strict connection to the occurrence dates and would not happen at any other time. For example, delay caused by typhoon cannot be assumed to happen at any other moment, but the real typhoon period. Another example maybe holiday or winter season, which also cannot be shifted.

- *Delay events with floating dates*, which have no strict link to calendar and presumably (but not necessary) would have occurred at any other moment of the project. As an example, there can be mistake in design, which required substantial changes, or inaccurate estimation of the activity duration, etc.

Major part of delay events fall in the first category. It is suggested in the present Study, that whenever it cannot be proved that delay event could have floating dates, it should be considered as a delay event with fixed dates and hence cannot be moved or shifted during delay analysis.

4.1.6 Mitigation and Acceleration as Delay Events

Thomas (2000) defined schedule acceleration as having more work to carry out in the same period of time or having a shorter period of time to carry out the same amount of work [**?**]. Acceleration of the remaining work of a project is typically ordered to compensate for the delays that occurred in the completed work of the project [6].

There are three types of acceleration: directed; constructive; and voluntary. A detailed discussion of each type can be found elsewhere (e.g., Jensen et al., 1997). Directed acceleration occurs when a contractor (or a subcontractor) is required by an owner (or a contractor) to perform the initial scope of work in a shorter amount of time than originally planned [31]. In terms of delay analysis the situation, when activity duration is actually shorter than was planned, becomes a point of interest.

Acceleration periods are often associated with extra-cost, which is are qualitatively different from delay damages and can itself become a source of dispute [58]. Same as delay events, acceleration periods fall all under the definition of "the difference between an intent and reality". They can create additional float, impact Project Completion date and even cause a disruption (through congestion on site, reduced productivity and an increase in defects, etc).[53]

Most of delay analysis methods only focus on the delayed activities and ignore the effect of time-shortened activity on the total project duration. [51] Failure to consider the acceleration periods is included in criticism of such conventional methods, as As-Planned Impacted or Collapsed As-Built.

Kim et. al (2005) recommended that delay analysis should simultaneously consider the effects of delayed and time-shortened activities. [56] In the present study it is suggested that for more accurate and reliable analysis, acceleration periods should be treated as delay events with the opposite sign, i.e. with negative duration $\omega_i^k = e_i^k - b_i^k < 0$

4.1.7 Consequences of Inaccurate planning

From a common sense point of view the term "delay event" suggests that there had been some happening or literally an "event", that has caused a project delay. However there can be situations when the "difference between plan and reality" lays in the defective planning and not in performance. In terms of timing three possible cases of inaccurate planning could be identified:

(i) incorrect estimation of activity duration;

(ii) need to drop or add some activity(s);

(iii) need of logic change (change in activities order, relations, constraints assignment, etc.).

It is suggested to use the concept of delay event to reflect the difference between original baseline and a new plan. Rise (or reduction) of activity duration, caused by inaccurate planning, should be depicted as delay event (usually certain) with 'floating' dates, which starts at the moment, when the necessity of extension became known.

Additional activity should be considered as delay event (which duration equals to the full length of new activity), and dropping activities - as acceleration (delay event with negative duration). During delay analysis these activities should be analyzed considering the timing, when they were actually introduced into the Schedule.

Logic change is more complicated issue. It can be a consequence of inaccurate planning or used as a tool to reduce Project duration during performance. Whatever the reason is, Logic change should be carefully considered. Next paragraph is devoted to more detailed description of how to deal with this type of project change.

4.1.8 Logic change

Logic change in conventional delay analysis methods can be handled in different ways:

- No consideration of logic change (API and CollAB group of methods)
- Logic change is applied within a window. Two variations are available:

- after analysis of one activity (or delay event) and before next;

- first update the window without logic change, get results, then - logic change;

- Logic change is applied not only within a window, but also for the rest of the project, but is not considered to be a delay event;
- Any logic change is considered as delay event (or acceleration);

4.1.9 'Pacing delays'

Another question, which challenges the But-For Schedules Analysis (and other DA methods) according to Zack (2001) is "would the project have been built at the same speed in the absence of these delays?". The author claims that when faced with a project delay (especially if it caused by the other party), contractors tend to slow down their other noncritical work.[107]

But when it comes to delay analysis, most available delay analysis methods ignore this delay type [51]. *Pacing delays* [106] are usually treated same as any other, and in comparison with concurrent delays may 'loose' the case. The analyst could draw a conclusion that if, for example, the owner-caused delay did not occur, the Project would have been anyway delayed by the Contractor's slow progress.

'Pacing delays' are connected to the *certain* delay events. Contractors would not slow down there work if it was not known at that time that parallel are activities would be delayed at least to some particular date. In other words, **'pacing delay' occur only in**

the parallel to the certain delay event. Both delay events in these case should be considered as having fixed dates.

4.1.10 Disruption

In terms of contract claims, disruption is an activity-specific loss of productivity caused by changes in the working conditions under which that activity was carried out [35]. Recommended Practice standard (AACE, 2011) defines disruptions "as an action or event which hinders a party from proceeding with the work or some portion of the work as planned or as scheduled."[1]

Disruption is not delay [82], but it can become a cause of delay to the Project completion. In the present study it is suggested that for the accuracy of delay analysis situations, when the project experiences disruption, should be described in terms of delay events, and all periods of delay, caused by disruption, increase in activities durations, periods of slow progress, etc. should be considered as delay events.

4.1.11 Serial and Independent Delay Events

Delay event can be caused by another delay event.

An independent delay is a particular delay which occurs in isolation or does not result from a previous delay and which effects can be readily calculated [8].

A delay which occurs solely as a consequence of an earlier, unrelated delay in the contract is called a serial delay [85]. Serial delays are sequences of successive nonoverlapping delays on a certain network path [8]. In addition, a serial delay may be caused by an independent delay [96].

In the present study it is suggested, that if particular delay event is a serial delay it should be notified (and labeled if necessary) prior to the main analysis.

Serial delay events should not be mistaken with simple shift of dates caused by the previous activity delay.

4.1.12 Extension of Time

It is provided in standard construction contracts, including FIDIC Red Book (1999), that for excusable delays the Contractor is entitled to Extension of Time (EOT). In the FIDIC Red Book (1999) clause 44.1 states, that "...the Engineer shall, after due consultation with the Employer and the Contractor, determine the amount of such extension and shall notify the Contractor accordingly, with a copy to Employer." [34]

EOT, given during the course of the project, *is* a Project change. Along with compensation of some previously encountered delays, it can create an additional float and become a causation of 'pacing delays'. For the accuracy of delay analysis it is suggested, that EOT should be carefully considered during delay analysis in terms, when it became known, was there any float created and who 'owns' it, were there any delays or disruptions as a result of it.

4.1.13 Criticality of Delay Events

It is generally believed that only delays laying on a critical path have impact on project completion date. For example, such conventional delay analysis technique as Collapsed As-built group of methods, are often criticized for inability to clearly identify the actual critical path. [53, 82] The reason lies in its changing-nature. During the course of the project, the schedule is constantly changes in time and logic, activities gain and loose float. What was critical at one point became non-critical at another and vice versa.

Conventional Delay Analysis methods can be distinguish in terms whether they require all delay events to be analyzed or only those, which belong to critical path. For example, Mohan et.al (2006) describing the differences between As-Planned Technique and As-Planned Impacted, shows that the former incorporates into schedule all delay events, and the latter first determines the critical path and then injects delay event to as-planned critical path.[73] This characteristic also distinguish Impacted As-Planned Technique in interpretation by Mohan et.al (2006) and As-Planned Impact Method, described by T. Farrow (2007), where the author suggests that all activities and all delay events in the project should be analyzed. [33, 73]

'Critical path delays' approach is attractive by its seeming simplicity, however in the project changing environment an assumption, that criticality of delay events can be clearly identified from the beginning of analysis, is perhaps too naive. Moreover, the fact of belonging to critical path does not automatically imply, that this delay event has caused delay to the project completion date. Activities 'compete' with each other over the project float. And that process should be investigated during delay analysis.

In the present study it is suggested that during delay analysis there should be considered *all delay events, encountered during the course of the project* , regardless to there position in as-planned or as-built critical path.

4.1.14 Scale of Delay Events

Naturally, the more parameters are considered during analysis, the more accurate and reliable results it provides. But in the mean time it makes analysis more complicated, expensive and time-consuming. And one of the major contributions to delay analysis heaviness is made by the chosen analysis scale.

Analogically to the project planning, which can be preformed at different levels from major sections and constructions to daily tasks, delay analysis can deal with months, weeks or even hours delays. Activities are not executed linearly, and small delay and acceleration periods can in fact occur within a single day or even an hour. Obviously, analysis of delays on daily basis is more rigorous than month by month. Higher scale makes results more approximate. But since in practice delay analysis is restricted with available data, the most reasonable solution lies in the correlation with project programmes and records.

In the present study is suggested, that prior to delay analysis the chosen scale of analyzed events should be carefully considered. Identified delay events on corresponding activities should be comparable with perceivable degree of approximation.

4.2 Other elements of Delay Analysis

4.2.1 Choice of the set of Delay Events

Delay analyst may choose delay events which belong to only one party or to all parties or in any kind of combination. Examples from Alkass et. all (1996) :

- All parties: Snapshot technique analysis
- One partys: But-For or collapsing technique
- One party plus neutral delay events: Isolated Delay Type (IDT)

4.2.2 Grouping of Delay Events

In conventional delay analysis methods delay events may be examined individually one-by-one or together as a group. Mohan and Al-Gahtani (2006) provide two implementations of the But-For or Collapsing technique technique: Unit Measure and Gross Measure.

Unit Measure describes an individual delay event analysis, when each delay event is added (subtracted) separately and then delay to the project caused by this particular delay event is calculated (Fig. 4.4 provides an example, when only owner-caused delay event was subtracted from the schedule).

Gross Measure corresponds with analysis, when several delays (for example, responsibility of one party) are added (subtracted) at one time. It can be in the form of one single delay event or separate events, but analyzed at the same time. Delay to the project in this case is a one number calculated for this group of delay events. (Fig. 4.5 illustrates an example, when all excusable delays were subtracted).

Delay events may be grouped by different criteria, such as its causation (responsible party), but also it may be defined by DA method's mechanic. For example, depending on if the technique is using 'window' approach, delay events delay events and its parts (all or responsibility of one party), which happen within the same window, could added (subtracted) together at one time. Delay to the project in this case is calculated for the analyzed window. The analyst could draw a conclusion, that in particular time period there occurred a certain amount of delay to the project completion date. How to apportion this amount of delay among the parties is another question.

Delay events occurred on one activity can be analyzed as a single delay, i.e. delay events are grouped by activity. For example in Project Games approach [30] delay to the project completion date is considered to be a contribution of certain delayed activities. In situations, when several delay events caused by different parties happened on the same activity, separate investigation would be needed.

There can be a medium approach, which combines both ways described above. In this case all delay events on one activity are analyzed as a single delay, which however is "cut" by windows into parts. Delay to completion is calculated for activity within time periods (ex. Time Impact Technique by Alkass et.all (1996)[4]).

Grouping of delay events makes an analysis faster, however in a pursuit of time (and cost) efficiency lies a risk of loosing accuracy and reliability of the analysis. It is suggested that whenever is possible, delay analyst should consider delay events individually.

4.2.3 Assessment of Delay to Completion: Straight or but-for logic

Delay to completion date is an adverse effect on the date, by which Contractor is contractually obliged to complete the works.[82] Usually it is a result of multiple delay events, and the main task which lies in front of the analyst is to allocate the responsibility for this delay among delay events (and hence among the parties responsible for these events).

In conventional delay analysis methods, using additive or subtractive approaches, the responsibility of analyzed delay event is assessed through calculation of the difference between the finish date of the programme, chosen as a baseline for analysis, and finish date of the entitlement schedule, achieved during analysis (ex. after adding or subtracting delay event, updating window, etc.)

The difference between two finish dates may be interpreted in two ways.

(1) By applying *Straight logic:*

If adding (subtracting) delay event has caused the Project finish date to become later (earlier), the resultant resultant amount of delay or time difference is assigned to this delay event (ex. As-Planned Impact Method by T. Farrow (2007)[33]).

(2) By applying *But-for logic:*

This approach usually applies when only one party's delay events are analyzed. Delay events are subtracted from (or added to) the program, and then the difference between actual as-built finish date and new finish date is calculated. The conclusion can be, that the Project might have finished earlier (or later) but-for other delay events and mitigation efforts, which were not analyzed. But-for logic focuses on delay events caused by the other party. In other words, if the Contractor is preparing delay analysis, the Owner's delays are analyzed and vice versa.

But-for logic in its essence contains assumptions. It provides an answer on what *might* have been to the Project, if some events were nor encountered. This kind of analysis may be helpful in assessing potential situations or when the project data is not sufficient for delay analysis, and there is no other way to allocated delay. However under assumption that all information is available and consistent, it is suggested that by definition straight logic is more appropriate for delay analysis.

4.2.4 Float Allocation

There are various ways to allocate float ownership among the parties. It can belong to the Project [90], to particular party (For example to the Contractor [91], and as some researchers suggest can be 'sold' to the Owner [37]) or shared according to some rule [46, 85, 81]

The discussion on how the float should be distributed is beyond the scope of present study.

Delay analysis, as a tool for supporting real project situations and disputes, ideally should be flexible and capable to deal with any kind of float distribution. Float allocation rule should become one of the inputs considered and chosen prior to analysis.

It is desirable, that every time, when delay event is introduced to the schedule, the consequential float change of activities is recorded. So that when requested an analyst is able to attribute float consumption to particular delay event. And then a question whether this amount of float was 'allowed' to be consumed by this delay event or not becomes a point of separate discussion.

4.2.5 Order of Addition/Subtraction of Delay Events

Most of conventional delay analysis methods use adding of events to the schedule as the main operation. The order in which delay events are added plays here a critical role and, as one can say, defines the results of the whole analysis. It also depends on the way, how delay events are chosen, if they grouped or considered individually, if the window concept is applied and how the delay to the project completion date is going to be assigned.

The order of delay events addition can be defined:

- Chronologically (from the beginning of the project) Delay events (or its parts) are added in chronological order by their start dates.
- By the type of delay event:
    - For example, first all excusable, then non-excusable, etc.
    - First all other party's delay events within a window, then neutral delay events as update before next window analysis (ex. Isolated Delay Type by Alkass et.all (1996)).

  - First certain, then uncertain, and many other ways

- By method requirements or analyst's decision: For example, isolated addition, described by Alkass et.all (1996) in the Time Impact Technique [4], when first the delay event is added to the schedule and after its impact is analyzed, it is removed, and the same is repeated for next delay event.

The order does not matter if:

- All analyzed delay events are added at the same time (grouped);

- All delay events (and parts) within one window are added at the same time;

Subtraction of delays can be done in different order and unlike with addition, where analyst always moves forward, subtraction can start from the end and go backwards, or from the first delay event and also move forward. Same as with addition, the order can be defined chronologically, by type of delay event or based on the analyst decision. If all delay events are subtracted at the same time (grouped), the order does not matter.

4.2.6 'Windows' and Window Selection Criteria

Window-based delay analysis methods perform delay analysis according to some extracted time frames, called windows.[51] These type of methods have been recognised as the most creditable methods for delay analysis. [40, 51, 56] And the choice of analyzed time period plays here a critical role.

Basically, there exist only two ways to select a 'window':

- When it is a subjective decision (based on project situation, major milestones, group of delays, etc. but still upon the analyst's choice);

- and when time periods are fixed. For example:
  - Window starts at the beginning of delay event or delaying activity (ex. Time Impact Technique by Alkass et.al (1996)[4]);
  - Time period equals one day (ex. Total float Management Technique by Mohan et.al (2006)[73]);
  - Delay periods are defined by overlapping of delay events and the project duration is divided to periods with no-delay, single-delay, and concurrent delay periods (as was suggested in DAMUDS by Y.Kim et all (2005)[56])

4.3 Project Games

Estévez-Fernández et al. applied cooperative game theory and serial cost sharing mechanisms to solve the problem of sharing responsibility of the delay (and costs associated with delay) [29, 30]. This approach assumes that delay to project completion is a responsibility of particular activity, which is the part of activity path(s).

Proposed algorithm is:

- First allocate the reward associated to the delay or expedition of each path (i.e. the total delay or total expedition created by its activities) among its activities.
- In a second step, use cooperative games to share the reward of the project among all its activities, using the initial allocations as reference points.

In terms of cooperative game theory, activity is treated as an agent, and activity path or any other group of activities (for example, responsibility of particular party) is treated as a coalition of agents (or activities).

Estévez-Fernández et al. (2009) showed, that the core of corresponding Project Game is nonempty and can be calculated by formula 4.1.

$$Core(v) = \{x \in R^N | \sum_{i \in N} x_i = v(N), \sum_{i \in S} x_i \geq v(S) forall S \in 2^N\} \quad (4.1)$$

where $v(S)$ represents the value of coalition (i.e. group of activities)

There are two major limitations of the Project Games approach:

- Set of activities are treated as coalitions without considering relationships within the project network;
- It is unable to address situations and distribute liability, when two or more different delay events occurred on the same activity.

However the Project Games concepts defines the maximum amount of delay each activity can be held responsible of. And hence conventional delay analysis methods can be evaluated by this criteria. Such as if some method assigns to activity responsibility for the delay more than its possible maximum, these results should be considered incorrect.

Prior to Analysis	
Delay Event	ω^k
Related Activity	i
Beginning date	b_i^k
Ending date	e_i^k
Duration	$e_i^k - b_i^k$
Responsible Party (Owner, Contractor, Neutral)	C / O/ N
Notification moment	t^k
Certainty (Yes or No)	$\circ$ / $\times$
With fixed dates (Yes or No)	$\circ$ / $\times$
Independence (Yes or No)	$\circ$ / $\times$
Causation	(Addition or subtraction of activity, result of Logic change, underestimated activity duration, pacing delay parallel to delay event ω^*, disruption period caused by delay event $\omega^{*}*$, Acceleration/mitigation efforts, etc.)
During Analysis	
Delay to Completion	D^k
Influence on Total Float of all affected activities	$\Delta TF_i = TF_i^k - TF_i^{k-1}$, $\Delta TF_j = TF_j^k - TF_j^{k-1}$, etc.

Table 4.1: Properties of Delay Events

4.4 Summary

Without exaggeration it can be said, that first of all results of delay analysis are determined by the choice and definition of delay events. Considering all points discussed above, it is suggested that prior and during delay analysis analysis for each delay event the following properties should be identified and assigned (Tab. 4.1).

One of the reasons why Delay Analysis is fraud with manipulations is the existence of so-called 'bottle necks' or weak points, where the analyst potentially can make a decision which determine and changes the results of the whole analysis. Definition and consideration of different types of delay events is a foundation stone, which alone is very important, but no enough to get a reliable delay analysis.

Table 4.2 summarizes the suggested decisions for the fragile spots of delay analysis structure.

Another important criteria is that delay for which an activity is responsible (as a result of delay analysis) is within a core of corresponding Project game.

Name of the Element	Possible Options	Desirable Solution
Set of Delay Events	One Party, All Parties, Combination	All Parties Delay Events
Group of Delay Events	Grouped by some criteria or Individual	Individual
Delay to Completion Assessment Logic	Staright or But-For	Staright Logic
Float Allocation		Float consumption should be recorded
Order of Addition/Subtraction of Delay Events	Chronological, by type of delay event, by method's requirement	Chronological
Window Selection Criteria	Subjective or fixed in DAM	Daily

Table 4.2: 'Bottle necks' of DAMs and its solutions

Chapter 5

FORMAL STRUCTURE OF DELAY ANALYSIS PROBLEM

5.1 Introduction

Delay is quite common in construction projects. Delay in completion of a project may bring siginificant losses on society. Delay can be caused by the Employer, the Conctractor, the third party or the force majeure that the parties to a contract cannot control. The liability of delay in completion of project should be appropriately allocated among the parties. In reality, the allocation of liability of project delay is the most likely source of contractual dispute in construction projects. As is mentioned in the earlier chapters, various types of methods to determine the allocation of liability of project delay. However, since there is no consensus on the most reasonable method, the parties to a contract often claims the analytical result which is derived by a delay analysis method that is likely to bring result more preferable for their own interest. Without the advance agreement on the employed delay analysis method, it is difficult to reach to a compromise by their voluntary negotiation.

Apart from such a practical problem, given the various types of delay analysis method that have been proposed so far, the question 'whether those delay analysis methods have any tendency in the analyzed results or not'. Another concern is that the allocation derived by those methods are fair or not.The first question concerns the comparative analysis of various delay analysis methods in terms of the tendency of analytical result. In order to derive a general conclusion, a general framework of the problem of allocating the liability of project delay must be developed.

The second question requires a conceptual framework to define '*fairness*' of allocation. Stimulated by the similar motivation to ours, Branzei *et al.*[18] proposes an approach based on cooperative game theory for determining fair shares for each agents who contributes to the delay of the project. It employs the concept of core and sharpley value for the fair allocation of delay cost. The cost sharing within the core satisfies at least guarantees that each agent bears the cost not more than his contribution to the cost of project delay. In the cooperative game theory framework, the contribution of each agent's delay to the delay cost of the project is defined by the definition of characteristic function. The cooperative game theory is a methodological framework to analyze the cooperative act of players with a common purpose, but being in conflict in allocating the gained fruit. Our study also employs the concept of cooperative game theory to discuss the fairness of allocation of liability of project delay. However, as is mentioned later, the difference of our work and the Branzei *et al.*[18] is that ours considers the network structure of project program whereas Branzei *et al.*[18] does not. Therefore, their study does not deal with the assessment of delay analysis methods which are commonly used in practice. Our study is rather interested in delay analysis methods that are commonly used in practice.

This chapter is organized as follows, In **5.2**, the formulation of project delay problem with consideration of the network structure of project program is proposed based on PERT (Project Evaluation and Review Technique) and CPM (Critical Path Method). In section **5.3**, the game of allocation of delay liability is formulated based on cooperative game theory concept. Based on the proposed game, the existence of core of the game is proved. Section **5.4** concludes the chapter and notes the remaining issues for the future studies.

5.2 Delay Analysis Method in Network

5.2.1 Analysis of PERT network

The structure of project program described by the PERT network is configurated by a list of activities and their sequence. Let $N = \{1, 2, \dots, n\}$ be the set of activities. The sequence of activities is defined by a list of predecessor activities and successor activities. Given a PERT network and a list of the time required to accomplish a task for each activity, the time required to complete the whole project is systematically calculated. Let b_i and e_i be the beginning date and the ending date of activity i. The date of commencement B and the date of completion E of the project with program Γ is

$$B = \min\{b_i, i \in N\}$$
$$E = \max\{e_i, i \in N\} \tag{5.1}$$

The duration of the project with a program is uniquely defined as

$$T = E - B \tag{5.2}$$

In addition, the amount of time that an activity can be delayed without causing the a delay in the completion date of the project is uniquely calculated. It is called as the *total float*.

5.2.2 Delay event

A delay event is the primary source of a delay in project completion. A delay event may change the ending date of project completion in two ways. One is that a delay event requires the revision of construction technique to be employed in the project, and hence, restructuring the project program. The other way is that a delay event requires longer time to complete an activity or activities than initially planned, but the structure of program network is maintained. The analytical method that we propose in this study is applicable to the latter type of delay event as we assume that delay events have no impact on the structure of program network.

The primary impact of any delay events in the model comes down to the additional time necessary to complete a task of an activity. Let $\Omega = \{\omega_1, \ldots, \omega_n\}$ be the set of delay events, where ω_i denotes delay event on activity i. Let $(d_1, \ldots, d_n)$ be the vector of additional time required to complete each activity. If no delay event occurs on activity i, ω_i is seen as a dummy delay event with $d_i = 0$.

5.2.3 Assessing the impact of delay event

A critical concern of the delay analysis is the assessment of the impact of delay events on the change of completion date of the whole project. The impact of a delay event is assessed by the 'what-if' principle, i.e., assessing the difference of completion date between the case if it occurs and the case if not, keeping the necessary time for the remaining activities same. The necessary time for the remaining activities can vary depending on which delay events have already impacted on the program. Assume that the part of delay events has already been impacted on the program. For a subset of delay events $S \subseteq \Omega \backslash \{\omega_i\}$, The impact of delay event ω_i on the change of completion date of the project is calculated as

$$D(\{\omega_i\}; S) \equiv E(S \cup \{\omega_i\}) - E(S), \tag{5.3}$$

where $E(S)$ denotes the completion date of the program with all delay events in S been impacted (hereinafter called as the program impacted by S). The project program before the delay event to be assessed is impacted is interpreted as a baseline to assess the impact of delay event on the completion date of the project. We call such a program as a baseline program. Delay impact is not necessarily defined only for single delay event. In more general way, delay impact of a set of delay events $T \subset \Omega$ with the baseline program impacted by $S,\ S \cap T = \phi$ been impacted is calculated as

$$D(T, S) \equiv E(T \cup S) - E(S).$$

5.2.4 Allocation

An allocation x assigns the total delay of whole project to each activity. An allocation is feasible iff

$$\sum_{i \in N} x_i \geq= D(\Omega, \phi) = \hat{D}. \tag{5.4}$$

This means that the allocation of liability as a solution of the probllem must cover at least the total delay of the whole project.

5.2.5 Float as rival goods

Let $TF_i(S)$ be the total float of activity i in the program impacted by S. The total float is the redundant duration defined on an activity on which delay event does not change the date of project completion. Total float of each activity is uniquely defined from the network structure of program. From the definition of total float, the impact of delay event ω_i on the activity i is written as

$$D(\{\omega_i\}; S) \equiv \max\{d_i - TF_i(S), 0\} \tag{5.5}$$

If activity j is a precedessor activity i, the delay of activity j directly affects on the beginning time of activity i. Hence, the consumption of total float due to the delay of acvitity j obviously reduce the available total float for activity i. Let $\{-i\}$ be the set of activities preceding to activity i. For any activity $j \in \{-i\}$,

$$\begin{aligned} &TF_i(S) = \max\{TF_i(S \cup \{\omega_j\}) - d_j, 0\} \\ &TF_j(S) = \max\{TF_j(S \cup \{\omega_i\}) - d_i, 0\} \\ &\text{for } S \subseteq \Omega \backslash \{\omega_i, \omega_j\} \end{aligned}$$

is satisfied. $TF_i(S \cup \{\omega_j\})$ is the amount of total float of activitity i assuming that delay event ω_j has not occurred, while $TF_i(S \cup \{\omega_j\})$ is that assuming delay event ω_j has occurred. Obviously,

$$TF_i(S) \geq TF_i(S \cup \{\omega_j\}) \tag{5.6}$$

$$TF_j(S) \geq TF_j(S \cup \{\omega_i\}) \tag{5.7}$$

is satisfied, total float is seen as a rival good for activity i and j in the sense that the consumption of total float due the the delay of activity i (activity j) reduces the amount of available total float to activity j (activity i).

Then, we investigate how the choice of baseline program makes a difference of evaluated delay impact of delay event. The assessments of the delay impact of ω_i in two different methods are compared to see how the choice of baseline program affects on the result of delay impact assessment. One is the assessment with the baseline program impacted by $S \subseteq \Omega \backslash \{\omega_i, \omega_j\}$ (i.e, $D(\{\omega_i\}; S)$) and the other is with the baseline program impacted by $S \cup \{\omega_j\}$ (i.e, $D(\{\omega_i\}; S \cup \{\omega_j\})$). From (5.6),

$$\begin{aligned} D(\{\omega_i\}; S) &= \max\{d_i - TF_i(S), 0\} \\ &\geq \max\{d_i - TF_i(S \cup \{\omega_j\}), 0\} \end{aligned}$$

$$= D(\{\omega_i\}; S \cup \{\omega_j\})$$

is derived. Likewise,

$$D(\{\omega_j\}; S) \geq D(\{\omega_j\}; S \cup \{\omega_i\})$$

Hence, the delay impact of delay event is assessed larger as more delay events on precedessor or follower activities are impacted on the baseline program. As is mentioned earlier, total float is a rival good. Inclusion of precedessor and follower activities of a delay event into the baseline program means that the total float has already owned and consumed by those activities. More generally, the following **Proposition 1** always holds.

Proposition 1: Let T be the set of activities on a path $\mathcal{P}$. If $T_1 = T \backslash T_2$ and $T_3 \subset T_4 \subseteq T_2$,

$$D(T_1; S \cup T_3) \leq D(T_1; S \cup T_4)$$

for any $S \subseteq \Omega \backslash T$.

Proposition 1 means that the function of delay impact $D(T_1; \cdot)$ is *superadditive*. The superadditivity implies that any party would always prefer the method of delay assessment based on the baseline program with less delay events of precedessor and follower activities impacted. The next subsection examines how the superadditivity creates conflict of interest of contractual parties in choosing the baseline program, hence the rule of delay analysis.

5.2.6 Float ownership

The choice of baseline program is obviously a critical concern for the parties to a contract in assessing the delay impact of delay event or the set of delay events. The choice of baseline program implicitly assumes 'who owns the float'. Consider the joint delay impact of delay event ω_i and ω_j to break up into the delay impact of each activity.

$$\begin{aligned} & D(\{\omega_i, \omega_j\}; S) \\ = \; & E(S \cup \{\omega_i, \omega_j\}) - E(S) \\ = \; & [E(S \cup \{\omega_i, \omega_j\}) - E(S \cup \omega_j)] \end{aligned}$$

$$
\begin{aligned}
&+ \quad [E(S \cup \omega_j) - E(S)] \\
&= \quad D(\{\omega_i\}; S \cup \{\omega_j\}) + D(\{\omega_j\}; S) \qquad (5.8)
\end{aligned}
$$

The above separation is assuming the total float is owned by activity j. The next separation is assuming that the total float is owned by activity i.

$$
\begin{aligned}
& \quad D(\{\omega_i, \omega_j\}; S) \\
&= \quad E(S \cup \{\omega_i, \omega_j\}) - E(S) \\
&= \quad [E(S \cup \{\omega_i, \omega_j\}) - E(S \cup \omega_i)] \\
&+ \quad [E(S \cup \omega_i) - E(S)] \\
&= \quad D(\{\omega_j\}; S \cup \{\omega_i\}) + D(\{\omega_i\}; S) \qquad (5.9)
\end{aligned}
$$

If both activities claims the ownership of total float, the assessment of delay impact of the two activities

$$
D(\{\omega_i\}; S) + D(\{\omega_j\}; S) \leq D(\{\omega_i, \omega_j\}; S), \qquad (5.10)
$$

which is not feasible allocation. Hence, if we apply such methods that the joint delay impact is broken down, it is necessary to determine a rule of 'who owns the float'. There are three rules in practice; 1) the Contractor owns, 2) the Employer owns and 3) the Project owns. If the rule 'the Contractor owns the float', the delay impact of Contractor's delay events is assessed based on the baseline program before the Employer's delay events are impacted. The delay impact of Employer's delay events is assessed based on the baseline program after the Contractor's delay events are impacted. If the rule 'the Employer owns the float' is choosen, the step is same with the Contractor owns rule, but the order of baseline program update is opposite. 'The Project owns the float' rule is the idea that the float ownership is determined on the first-come and first-served basis.

5.2.7 Concurrency

Concurrecy is defined as the occurence of two or more delay events at the same time, one an Employer risk event, the other a Contractor risk event. When the effects of both events have a direct impact on the completion date and are experienced at the same time, this is a 'concurrenct delay.' When two 'concurrent delays' arise at different times, but the effects are felt (in who or in part) at the same time, this is also known as 'concurrent effect.'[53] When multiple delay events occur at the same time, they contribute to the delay of whole project

For example, consider a very simple activity graph with two activities. Two activities are parallel and requires same period of time for completion. If both of activities are implacted by 5 days of delay events at the same time, the completion date of the project will be delayed by 5 days. Obviously, both of activities contribute to the delay of whole project in common. Even if either of two activities did not occur, it would not mean that the delay of whole project changes. The concurrency of delay events means that the impact of a delay event is offset by the delay of other delay events that occur at the same time. Here, we try providing a formal definition of concurrent delay.

Let C_p be a set of activities included in a feasible cut of the activity graph. $T = \{\omega_i | i \in C_p\}$ is the set of delay events on which activities are C_p.

Definition (Concurrent delay): Consider a subset of delay event $S \subset \Omega \backslash T$. If more than one delay event in T satisfies

$$i \in CL_p(S \cup \{\omega_i\}), \tag{5.11}$$

those delay events are concurrent delay. Here $CL_p(\cdot)$ denotes a set of activities in C_p with 'zero' total float. Hence (5.11) means that activity i is on critical path of project program with the delay events in $S \cup \{\omega_i\}$ impacted.

Based on the above-mentioned definition of concurrent delay, we demonstrate how a concurrent delay causes a conflictive preference of contractual parties over the choice of baseline program. The most important mechanism regarding the concurrent delay is that a delay event on one path creates additional total float of activities on other paths. Then, the additional total float which has a critical effect on the evaluation of the delay impact of delay events. In formal, if activity i and acitivity j is not in a procedessor-follower relationship, the following equations are always satisfied.

$$\begin{aligned} & TF_i(S \cup \{\omega_j\}) - TF_i(S) \\ = \;& \max\{d_j - TF_j(S), 0\} \qquad (5.12) \\ & TF_j(S \cup \{\omega_i\}) - TF_j(S) \\ = \;& \max\{d_i - TF_i(S), 0\} \qquad (5.13) \\ & \text{for any } S \subseteq \Omega \backslash \{\omega_i, \omega_j\} \end{aligned}$$

Then it implies that the total float based on the the baseline program with consideration of concurrent delay is always longer than the total float without consideration of concurrent delay.

The critical interest here is to see how a concurrent delay makes a difference in the evaluation of delay impact depending on the choice of baseline program. Considering the relationship of total float depending on the choice of baseline program, the assessment of delay impact of concurrent delay can be derived as follows. (See A)

Proposition 2: For any delay event ω_i and ω_j that satisfies (5.11), the assessment of delay impact of delay event ω_i and ω_j satisfies the inequality

$$D(\{\omega_j\}; S \cup \{\omega_i\}) \leq D(\{\omega_j\}; S) \qquad (5.14)$$

$$D(\{\omega_i\}; S \cup \{\omega_j\}) \leq D(\{\omega_i\}; S) \qquad (5.15)$$

$$\text{for any } S \subseteq \Omega \backslash \{\omega_i, \omega_j\}.$$

In terms of mathematical system, **Proposition 2** means that the delay impact function $D(\{\omega_i\}, \cdot)$ is *subadditive* for concurrency delay. The contrast with the case of competitive float is interesting. For the concurrent delay, any party to a contract would prefer the baseline program with concurrent delay event impacted. Again, this is because the delay of a path creates a new total float for the activities on other paths. Then every party would wish to be entitled to use the newly created float.

5.3 A Game for Delay Liability Allocation

5.3.1 Cooperative game and project game

Cooperative game theory is an analytical framework to investigate the formation of strategic cooperative coalition and its feasible allocation of payoff in the solution. Cooperative game theory defines a 'characteristic function' of the game, say $v(S)$, which represents the total amount of transferable utility that the members of coalition S could earn without any help from the players outside of S.

Employing the concept of cooperative game theory to apply for the allocation of delay liability, we define the following characteristic function.

$$v(T) = \max_{S \subseteq \Omega \setminus T} D(T; S). \tag{5.16}$$

$v(T)$ represent the maximum assessment of delay impact of delay events in T. If they are allocated the liability more than $v(T)$, they can claim against the court that such an allocation is not fair. This notion corresponds to the feature of the characteristic function in cooperative game theory in that the amount of a coalition is that the minimum payoff for the members of the coalition guaranteed by the outside option. Given the characteristic funtion defined as (5.16), **Proposition 3** is derived.

Proposition 3: Given the definition of characteristic function (5.16), the coalitional game (N, v) is a concave game, i.e.

$$v(T \cup \{i\}) - v(T) \geq v(S \cup \{i\}) - v(S) \tag{5.17}$$

$$\text{for all } T \subset S \text{ and } j \in N \setminus S.$$

Proposition 3 suggests that the problem of delay liability allocation is a concave game, hence it implies the existence of core defined as follows.

An allocation x is in the core of v iff x is feasible and no coalition can improve x.

$$\sum_{i \in N} x_i = D \text{ and } \sum_{i \in T} x_i \leq v(T), \ \forall T \subseteq N \tag{5.18}$$

5.4 Conclusion

Amid the existence of various delay analysis methods, two questions have remained to be unanswered, i.e., the general tendency of analytical results of various delay analysis methods and the evaluation of those analytical results in terms of 'fairness'. In order to tackle those questions, this chapter aimed to developing a formal model of the problem on the allocation of delay liability in a construction project to systematically understand the mechanism of a delay event has an impact on the delay of the whole project in the network of project program. The first remark is that the source of delay program arises from the existence of float. Float cannot be shared among activities on a same path. If all parties claims the ownership of total float, the sum of allocated liability does not cover the total delay of the project. Float matters in terms of two aspects, i.e. the float ownership and the concurrent delay. If the parties to a contract agree on the rule to determine the rules for the float ownership and the concurrent delay in advance, the allocation of delay liability can be uniquely determined. The second remark is that the problem of fair solution of delay liability allocation has a structure common with cooperative game theory. In addition, the coalition game with the characteristic function defined by the possible maximum assessment of delay impact has a core. Then, it is proved that any problem of delay liability allocation has fair allocation.

Chapter 6

CONCLUSIONS

Nearly every construction project experiences time overrun beyond originally planned dates. Significant delays damage the project and often result in additional cost, payment withhold and demand for liquidated damages. Several studies have shown that contract schedule and payment issues are two most common items of dispute in construction.

Delay can be caused by the Employer, the Conctractor, the third party or the force majeure that the parties to a contract cannot control. The liability of delay in completion of project should be appropriately allocated among the parties.

In order to establish liability for the time and cost overrun all project records are gathered and carefully investigated. However knowing the facts, even if the information is complete and consistent, does not mean the responsibility can be immediately allocated among the parties. Project delay is complex in nature; it is caused by various happenings and factors influencing at different periods of time and attributed to different parties. To establish the liability further investigation is required. In modern project management practice it is called Delay Analysis (DA).

Over past twenty years several DA methods or techniques have been created. Several studies have been devoted to the DA methods comparison and have proved that there is no consensus on the most reasonable method, application of two methods to the same project usually results in different if not contrary conclusions. And this in turn instead of clarification just leads to a new argument which DA is more reliable and which technique is preferable. Without the advance agreement on the employed delay analysis method, it is difficult to reach to a compromise by their voluntary negotiation.

However behind that lies another fundamental problem confusion and lack of body of knowledge or standard in definition of delay concept itself and methods' algorithms. For example such well known method as TIA can be understood differently in terms what delay event is, in which order they should be added, etc. In fact most of DAs are not based on any methodology at all, but just on experts logic and experience. So the establishment of DA body of knowledge and application rules becomes an important subject of research.

Construction schedules can be very complex involving thousands of components. It is impossible to analyze it on "one screen" or display all components, but a mistake in one may result in a misleading conclusion. DA can be performed correctly only by a high skilled professional, and DA techniques are made as useful tool for assistance.

Apart from such a practical problem, given the various types of delay analysis method that have been proposed so far, the question 'whether those delay analysis methods have any tendency in the analyzed results or not'. Another concern is that the allocation derived by those methods are fair or not.The first question concerns the comparative analysis of various delay analysis methods in terms of the tendency of analytical result. In order to derive a general conclusion, a general framework of the problem of allocating the liability of project delay must be developed.

The present study reviews existing delay analysis methods, its interpretations, variations and assessment of strong and weak points by different researchers. It formalizes conventional delay analysis methods through developing of unified mathematical language and extends the concept of delay event, defines the desirable properties of delay events and evaluate existing DA techniques in terms of correspondence to them.

One of the reasons why Delay Analysis is fraud with manipulations was found in the existence of so-called 'bottle necks' or weak points, where the analyst potentially can make a decision which determine and changes the results of the whole analysis. The requirements to desirable delay analysis method were formulated.

Amid the existence of various delay analysis methods, two questions have remained to be unanswered, i.e., the general tendency of analytical results of various delay analysis methods and the evaluation of those analytical results in terms of 'fairness'. The first remark is that the source of delay program arises from the existence of float. Float cannot be shared among activities on a same path. If all parties claims the ownership of total float, the sum of allocated liability does not cover the total delay of the project. Float matters in terms of two aspects, i.e. the float ownership and the concurrent delay. If the parties to a contract agree on the rule to determine the rules for the float ownership and the concurrent delay in advance, the allocation of delay liability can be uniquely determined. The second remark is that the problem of fair solution of delay liability allocation has a structure common with cooperative game theory. In addition, the coalition game with the characteristic function defined by the possible maximum assessment of delay impact has a core. Then, it is proved that any problem of delay liability allocation has fair allocation.

Appendix A

Proofs

Proof of Proposition 2

For the initial setting of program with the delay events in S impacted, there are two different cases:

Case 1

$$TF_i(S) \geq 0 \text{ and } TF_j(S) = 0 \tag{A.1}$$

Case 2

$$TF_i(S) = 0 \text{ and } TF_j(S) \geq 0 \tag{A.2}$$

Case 1

Consider the delay impact of delay event ω_j with two baseline programs: one includes ω_i and the other does not. From (A.1) and (5.13),

$$TF_j(S \cup \{\omega_i\}) = d_i - TF_i(S) \geq 0$$

From the definition of delay impact (5.5) and (A.1), delay impact of ω_j with baseline program excluding ω_i is derived as

$$\begin{aligned} & D(\{\omega_j\}; S \cup \{\omega_i\}) \\ = \; & \max\{d_j - TF_j(S \cup \{\omega_i\}), 0\} \\ = \; & d_j - [d_i - TF_i(S)] \end{aligned}$$

On the other hand,

$$\begin{aligned} D(\{\omega_j\}; S) &= \max\{d_j - TF_j(S), 0\} \\ &= d_j \end{aligned} \tag{A.3}$$

Hence obviously,

$$D(\{\omega_j\}; S \cup \{\omega_i\}) \leq D(\{\omega_j\}; S) \tag{A.4}$$

is satisfied.

Consider the delay impact of delay event ω_i with two baseline programs: one includes ω_j and the other does not. From (A.1) and (5.12),

$$TF_i(S \cup \{\omega_i\}) - TF_i(S) = d_j$$

From the definition of delay impact (5.5) and (A.1), delay impact of ω_i with baseline program including ω_j is derived as

$$D(\{\omega_i\}; S) \;\; = \;\; \max\{d_i - TF_i(S), 0\} = d_i$$

On the other hand,

$$\begin{aligned} & D(\{\omega_i\}; S \cup \{\omega_j\}) \\ = \;& \max\{d_i - TF_i(S \cup \{\omega_j\}), 0\} \\ = \;& \max\{d_i - [d_j - TF_i(S \cup \{\omega_j\}], 0\} \\ = \;& d_j - [d_i - TF_i(S)] \end{aligned}$$

Hence obviously,

$$D(\{\omega_i\}; S \cup \{\omega_j\}) \leq D(\{\omega_i\}; S) \tag{A.5}$$

is satisfied.

Case 2

Consider the delay impact of delay event ω_i with two baseline programs: one includes

ω_j and the other does not. From (A.2) and (5.12),

$$TF_i(S \cup \{\omega_i\}) = d_j - TF_i(S) \geq 0$$

From the definition of delay impact (5.5) and (A.1), delay impact of ω_b with baseline program excluding ω_a is derived as

$$\begin{aligned} & D(\{\omega_i\}; S \cup \{\omega_j\}) \\ = \; & \max\{d_i - TF_i(S \cup \{\omega_j\}), 0\} \\ = \; & d_i - [d_j - TF_i(S)] \end{aligned}$$

On the other hand,

$$\begin{aligned} D(\{\omega_i\}; S) &= \max\{d_i - TF_i(S), 0\} \\ &= d_i \end{aligned} \tag{A.6}$$

Hence obviously,

$$D(\{\omega_i\}; S \cup \{\omega_j\}) \leq D(\{\omega_i\}; S) \tag{A.7}$$

is satisfied.

Consider the delay impact of delay event ω_j with two baseline programs: one includes ω_i and the other does not. From (A.2) and (5.13),

$$TF_j(S \cup \{\omega_i\}) - TF_j(S) = d_i$$

From the definition of delay impact (5.5) and (A.1), delay impact of ω_j with baseline program including ω_i is derived as

$$D(\{\omega_j\}; S) \;=\; \max\{d_j - TF_j(S), 0\} = d_j$$

On the other hand,

$$D(\{\omega_j\}; S \cup \{\omega_i\})$$

$$
\begin{aligned}
&= \max\{d_j - TF_j(S \cup \{\omega_i\}), 0\} \\
&= \max\{d_j - [d_i - TF_j(S \cup \{\omega_i\}], 0\} \\
&= d_i - [d_j - TF_j(S)]
\end{aligned}
$$

Hence obviously,

$$D(\{\omega_j\}; S \cup \{\omega_i\}) \leq D(\{\omega_j\}; S) \tag{A.8}$$

is satisfied. Then **Proposition 2** is proved.

Bibliography

[1] AACE International (2011). “Forensic Schedule Analysis TCM Framework 6.4 - Forensic Performance Assessment.” AACE International Recommended Practice No. 29R-03.

[2] Adhikari, I., Kim, S. Y., and Lee, Y. D. (2006). “Selection of Appropriate Schedule Delay Analysis Method: Analytical Hierarchy Process (AHP).” PICMET Proceedings, Istanbul, Turkey, 483-488.

[3] Aibinu, A. A., and Odeyinka, H. A. (2006). “Construction Delays and Their Causative Factors in Nigeria.” Journal of Construction Engineering and Management, ASCE, July 2006, 667-677.

[4] Alkass, S., Mazerolle, M., and Harris, F. (1996). “Construction delay analysis techniques.” Construction Managenemt and Economics 14, 375-394.

[5] Arcuri, F. J., and Hildreth, J. C. (2007). “The Principles of Schedule Impact Analysis.” Report for Virginia Department of Transportation, Virginia Polytechnic Institute and State University.

[6] Arditi, D. and Patel, B. K. (1989). “Impact analysis of owner-directed acceleration.” Journal of Construction Engineering and Management, ASCE, 115(1), 144-157.

[7] Arditi, D., and Pattanakitchamroon, T. (2006). “Selecting a delay analysis method in resolving construction claims.” International Journal of Project Management, 24, 145155.

[8] Arditi, D. and Robinson, M. A. (1995). Concurrent delays in construction litigation. Cost Engineering, AACE, 37(7), 20-30

[9] Assaf, S. A., and Al-Hejji, S. (2006). “Causes of delay in large construction projects.” International Journal of Project Management 24, 349357.

[10] Barraza, G. A., and Bueno, R. A. (2007). “Probabilistic Control of Project Performance Using Control Limit Curves.” Journal of Construction Engineering and Management, ASCE, December 2007, 957-965.

[11] Barry, D. (2009). “Beware the Dark Arts! Delay Analysis and the Problems with Reliance on Technology.” Society of Construction Law (UK).

[12] Bergantiños, G., and Sánchez, E. (2002). “How to Distribute Costs Associated with a Delayed Project.” Annals of Operations Research 109, 159174.

[13] Beşoğul E. (2006). “Reasons of delays in steel construction projects: an application of a delay analysis methodology.” Master Thesis, Grad.School of Natural and Applied Sciences, Middle East Technical University.

[14] Bordoli, D. W., and Baldwin A. N. (1998). “A methodology for assessing construction project delays.” Construction Management and Economics 16, 327-337.

[15] Borm, P., Hamers, H., and Hendrickx, R. (2001). “Operations Research Games: A Survey.” Top, vol. 9, no. 2, December 2001, 139-216.

[16] Braimah, N., and Ndekugri, I. (2008). “Factors influencing the selection of delay analysis methodologies.” International Journal of Project Management, 26, 789799.

[17] Braimah, N., and Ndekugri, I. (2009). “Consultants Perceptions on Construction Delay Analysis Methodologies.” Journal of Construction Engineering and Management, ASCE, December 2009, 1279-1288.

[18] Brânzei, R., Ferrari, G., Fragnelli, V., and Tijs, S. (2002). “Two Approaches to the Problem of Sharing Delay Costs in Joint Projects.” Annals of Operations Research 109, 359374.

[19] Baram, G. E. (2000). “Concurrent delays - what are they and how to deal with them?” AACE International Transactions, CDR.07.1-8.

[20] Bramble B.B., Callahan M.T. (1987), “Construction delay claims”. Wiley Law Publicaions, New York.

[21] Bramble B.B., Callahan M.T. (2000), “Construction delay claims”. Wiley Law Publicaions, New York.

[22] Bubshait, A. A., and Cunningham, M. J. (1998). "Comparison of Delay Analysis Methodologies." Journal of Construction Engineering and Management, July/August 1998, 315-322.

[23] Cambridge Advanced Learner's Dictionary & Thesaurus Cambridge University Press, 2011

[24] Castro, J., Gómez, D., and Tejada, J. (2007). "Aproject game for PERT networks." Operations Research Letters 35, 791 798.

[25] Critchlow, J., Farr, A., Briggs, S., Hammond, J., Pickavance, K., and Lavers, A. (2006). "The Great Delay Analysis Debate." Society of Construction Law.

[26] Cui, D., and Yao, L. (2009). "Calculation Method of Delay Claims Based on Shapley Value." Conference Paper, Industrial Engineering and Engineering Management, 420-423.

[27] Cushman, R. F., Cushman, K. M. and Cook, S. B. (1990). "Construction Litigation: Representing the Owner". John Wiley & Sons, NY.

[28] Diekmann, J. E. and Nelson, M. C. (1985). "Construction claims: frequency and severity." Journal of Construction Engineering and Management, ASCE, 111(1), 74-81.

[29] Estévez-Fernández, A., Borm, P., and Hamers, H. (2007). "Project games." International Journal on Game Theory, Vol.36, 149176.

[30] Estévez-Fernández, A. (2009). "A Game Theoretical Approach to Sharing Penalties and Rewards in Projects." Discussion Paper, Dept. of Econometrics and Operations Research, VU University Amsterdam, and Tinbergen Institute.

[31] Evans, M. E. (2004). "Time-related claims: the basics." Ohioconstructionlaw.com newsletter, 5(2), 2-7.

[32] Farrow, T. (2001). "Delay Analysis Methodology and Mythology." Society of Construction Law.

[33] Farrow, T. (2007). "Developments in the Analysis of Extensions of Time." Journal of Professional Issues in Engineering Education and Practice, ASCE, July 2007, 218-228.

[34] FIDIC Red Book, 1999

[35] Finke, M. R. (1999). "Window Analyses of Compensable Delays." Journal of Construction Engineering and Management, ASCE, March/April 1999, 96-100.

[36] Fong, N.K., Wong, L.Y., and Wong, L.T. (2005). "Fire services installation related contributors of construction delays." Building and Environment, 41, 211222.

[37] de la Garza, J. M., Prateapusanond, A., and Ambani. N. (2007). "Preallocation of Total Float in the Application of a Critical Path Method Based Construction Contract." Journal of Construction Engineering and Management, ASCE, November 2007, 836-845.

[38] Gauthier, D. (1993). "Fairness and Cores: A Comment on Laden." Philosophy & Public Affairs, vol. 22, no. 1, Winter 1993, 44-47.

[39] Gerk, J. E. V., and Qassim, R. Y. (2008). "Project Acceleration via Activity Crashing, Overlapping, and Substitution." IEEE Transactions on Engineering Management, vol. 55, no. 4, November 2008, 590-601.

[40] Gothand, K. D. (2003). "Schedule Delay Analysis: Modified Windows approach." Cost Engineering, vol. 49, no. 9, 18-23.

[41] Han, S. H. , Yun, S. , Kim, H. , Park, H. K. , and Lee, S. H. (2009). "Analyzing Schedule Delay of Mega Project." IEEE Transactions on Engineering Management, vol. 56, no. 2, 243-256.

[42] Hegazy, T., and Zhang, K. (2005). "Daily Windows Delay Analysis." Journal of Construction Engineering and Management, ASCE, May 2005, 505-512.

[43] Hegab, M. Y., and Smith, G. R. (2007). "Delay Time Analysis in Microtunneling Projects." Journal of Construction Engineering and Management, ASCE, February 2007, 191-195.

[44] Hegazy, T., and Menesi, W. (2008). "Delay Analysis under Multiple Baseline Updates." Journal of Construction Engineering and Management, ASCE, August 2008, 575-582.

[45] Hegazy, T., Said, M., and Kassab, M. (2011). "Incorporating rework into construction schedule analysis." Automation in Construction, Vol.20, 10511059.

[46] Householder, J.L., Ruthland H.E. (1990), "Who owns the float?", Journal of Construction Engineering and Management ASCE, 116 (1), 130 - 3.

[47] Ibbs, W., and Nguyen, Long D. (2007). "Schedule Analysis under the Effect of Resource Allocation." Journal of Construction Engineering and Management, ASCE, February 2007, 131-138.

[48] Ibbs, W., Nguyen, L. D., and Lee, S. (2007). "Quantified Impacts of Project Change." Journal of Professional Issues in Engineering Education and Practice, ASCE, January 2007, 45-52.

[49] Iyer, W., and Jha, K. N. (2006), "Critical factors affecting schedule performance: evidence from Indian construction projects", Journal of Construction Engineering and Management, 132(8), 871-881.

[50] James, D. W. (1991). "Concurrency and apportioning liability and damages in public contract adjudications." Public Contract Law Journal, 20(4), 490-531.

[51] Kao, C.-K., and Yang, J.-B. (2009). "Comparison of windows-based delay analysis methods." International Journal of Project Management 27, 408418.

[52] Kartam, S. (1999). "Generic Methodology for Analyzing Delay Claims." Journal of Construction Engineering and Management, ASCE, November/December 1999, 409-419.

[53] Keane, P.J., and Caletka, A.F. (2008). "Delay Analysis in Construction Contracts." Blackwell Publishing.

[54] Kim, C. S., and Liu, L. Y. (2007). "Cost Information Model for Managing Multiple Projects." Journal of Construction Engineering and Management, ASCE, December 2007, 966-974.

[55] Kim, K., and de la Garza, J. M. (2005). "Critical Path Method with Multiple Calendars." Journal of Construction Engineering and Management, ASCE, March 2005, 330-342.

[56] Kim, Y., Kim, K., and Shin, D. (2005). "Delay Analysis Method Using Delay Section." Journal of Construction Engineering and Management, ASCE, November 2005, 1115-1164.

[57] Koehn, E., Young, R., Kuchar, J., and Seling, F. (1978). "Cost of delays in construction." Journal of the Construction Division, 104(3), 323-331.

[58] Kutil, P. M. and Martin, M. F. (1995). "Constructive acceleration." Construction Briefings, 95-13.

[59] Kumaraswamy M.M., (1997) "Conflicts, claims and disputes in construction", Engineering, Construction and Architectural Management, Vol. 4 Iss: 2, 95 - 111

[60] Kumaraswamy M.M., (1998),"Tracing the roots of Construction Claims." COBRA 1998 Construction and Building Research Conference, v. 2, p. 26 - 35

[61] Laden, A. (1991). "Games, Fairness, and Rawls's A Theory of Justice." Philosophy & Public Affairs, vol. 20, no. 3, 189-222.

[62] Lee, H.-S. , Ryu, H.-G., Yu, J.-H., and Kim, J.-J. (2005). "Method for Calculating Schedule Delay Considering Lost Productivity." Journal of Construction Engineering and Management, ASCE, November 2005, 1147-1154.

[63] Lee, J.-S., and Diekmann, J.E. (2011). "Delay Analysis considering production rate." Canadian Journal of Civil Engineering, Vol.38, 361-372.

[64] Littlechild, S. C., and Owen, G. (1973). "A Simple Expression for the Shapely Value in a Special Case." Management Science, Vol. 20, No. 3, Theory Series, 370-372.

[65] Lo, T. Y., Fung, I. W. H., and Tung, K. C. F. (2006). "Construction Delays in Hong Kong Civil Engineering Projects." Journal of Construction Engineering and Management, ASCE, June 2006, 636-649.

[66] Love, M. K. (2000). "Theoretical delay and overhead damages." Public Contract Law Journal, 30(1), 33-64.

[67] Lovejoy, V.A. (2004), "Claims schedule development and analysis: collapsed as-built scheduling for beginners." Cost Engineering Journal, AACE 46(1), 27-30.

[68] Lowsley, S., and Linett, C. (2006) "About Time: Delay Analysis in Construction", RICS Business Servisec Ltd., London.

[69] Lucas, E. D. (2002). "Schedule Analyzer Pro - An Aid in the Analysis of Delay Time Impact Analysis." Cost Engineering, vol. 44, no. 8, August 2002, 30-36.

[70] Majid, A. and McCaffer, R. (1998). “Factors of non excusable delays that influence contractor’s performance”. Jornal on Management and Engineering, 14(3): 42-49.

[71] Marzouk, M., El-Dokhmasey, A., and El-Said, M. (2008). “Assessing Construction Engineering-Related Delays.” Journal of Professional Issues in Engineering Education and Practice, ASCE, July 2008, 315-326.

[72] Mbabazi, A., Hegazy, T., and Saccomanno, F. (2005). “Modified But-For Method for Delay Analysis.” Journal of Construction Engineering and Management, ASCE, October 2005, 1142-1144.

[73] Mohan, B., and Al-Gahtani, K. S. (2006). “Current Delay Analysis Techniques and Improvements.” Cost Engineering vol. 48, no. 9, September 2006, 12-21.

[74] Morelli, M., and Sacco, P. L. (1997). “Game-theoretic definitions of fairness and the contractualistic foundations of justice.” Research in Economics, Vol.51, 387-400.

[75] Moulin, H., and Shenker, S. (1992). “Serial Cost Sharing.” Econometrica, vol. 60, no. 5, 1009-1037.

[76] Nash, S. C (2002). “Delay and Disruption: Legal Considerations”. Online only, Society of Construction Law.

[77] Ndekugri, I., Braimah, N., and Gameson, R. (2008). “Delay Analysis within Construction Contracting Organizations.” Journal of Construction Engineering and Management, ASCE, September 2008, 692-700.

[78] Nguyen L.D. (2007). Doctoral Thesis, “The Dynamics of Float, Logic, Resource Allocation, and Delay Timing in Forensic Schedule Analysis and Construction Delay Claims”. University of California, Berkeley

[79] Nguyen, L. D., and Ibbs, W. (2008). “FLORA: New Forensic Schedule Analysis Technique.” Journal of Construction Engineering and Management, ASCE, July 2008, 483-491.

[80] Oliveros, A., and Fayek, A. (2005). “Fuzzy Logic Approach for Activity Delay Analysis and Schedule Updating.” Journal of Construction Engineering and Management, ASCE, 131(1), 42-51.

[81] Pasiphol S., Popescu C. (1994), "Qualitative criteria combination for total float distribution", ASCE International, DCL.3.1 - DCL.3.6.

[82] Keith Pickavance (2005), "Delay and Disruption in Construction Contracts", Third Edition.

[83] Peters, T. F. (2003). Dissecting the doctrine of concurrent delay. AACE International Transactions, CDR.01.1-8.

[84] Pinnell, S. S. (1998). "How to Get Paid for Construction Changes: Preparation and Resolution Tools and Techniques", McGraw-Hill, New York, NY.

[85] Ponce de Leon, G. (1987). Theories of concurrent delays. AACE Transactions. H.6.1-5

[86] Rubin, R. A., Guy, S. D., Maevis, A. C., and Fairweather, V. (1983). Construction Claims, Analysis, Presentation and Defense, Van Nostrand Reinhold, NY.

[87] Sakka, Z. I., and El-Sayegh, S. M. (2007). "Float Consumption Impact on Cost and Schedule in the Construction Industry ." Journal of Construction Engineering and Management, ASCE, February 2007, 124-130.

[88] Sandlin, L.S., Sapple, J. R., and Gautreaux, R.M. (2004). "Phased Root Cause Analysis A Distinctive View on Construction Claims." Cost Engineering vol. 46, no. 6, June 2004, 16-20.

[89] Schumacher, L. (1995). Quantifying and apportioning delay on construction projects. Cost Engineering, AACE, 37(2), 11-13.

[90] SCL (2002). "Delay and Disruption Protocol." Society of Construction Law.

[91] Scott, S., Harris, R. A., and Greenwood, D (2004). "Assessing the New United Kingdom Protocol for Dealing with Delay and Disruption." Journal of Professional Issues in Engineering Education and Practice, ASCE, January 2004, 50-59.

[92] Semple, C., Hartman, F. T. and Jergeas, G. (1994). "Construction claims and disputes: causes and cost/time overruns." Journal of Construction Engineering and Management, ASCE, 120(4), 785-795.

[93] Sgarlata M.A., Brasco C.J. (2004), "Successful claims resolution through an understanding of the law governing allocation of risk for delay and disruption". Journal of Construction Management, CMAA, http://www.cmaanet.org/ejournal.php

[94] Shapley L.S. (1965). "Cores of Convex Games." Conference Paper, Fifth Informal Conference on Game Theory, Princeton University, April 1965.

[95] Sprumont, Y. (1998). "Ordinal Cost Sharing." Journal of Economic Theory, 81, 126-162.

[96] Stumpf, G. R. (2000). "Schedule delay analysis." Cost Engineering, vol. 42, no. 7, 32-43.

[97] Sweis, G., Sweis, R., Hammad A. A., and Shboul A. (2008). "Delays in construction projects: The case of Jordan." International Journal of Project Management, 26, 665674.

[98] Tieder, J. B. (2009). "Methods of Delay Analysis and How They are Viewed by the United States Legal System." Conference Paper, Society of Construction Law.

[99] Trauner, T. J. (1990), Construction Delays: Documenting Causes, Winning Claims, Recovering Costs, R.S. Means, Kingston, MA.

[100] Tserng, H. P., and Teng, W.-K. (2009). "Analysing dispute mediation cases of infrastructure projects through project life cycle." Structure and Infrastructure Engineering Vol. 5, No. 6, 515531.

[101] Merriam-Webster Online Dictionary, http://www.merriam-webster.com

[102] Yang, J.-B., Kao, C.-K., and Lee, Y.-Y. (2006). "System Requirement Analysis of a Construction Delay Analysis System." Conference Paper, The 23rd International Symposium on Automation and Robotics in Construction, 102-106.

[103] Yang, J.-B., and Yin, P.-C. (2009). "Isolated Collapsed But-For Delay Analysis Methodology." Journal of Construction Engineering and Management, ASCE, July 2009, 570-578.

[104] Yang, J.-B., and Kao, C.-K. (2012). "Critical path effect based delay analysis method for construction projects." International Journal of Project Management, Vol.30, Issue 3, 385397.

[105] Yates, J. K., and Epstein, A. (2006). "Avoiding and Minimizing Construction Delay Claim Disputes in Relational Contracting." Journal of Professional Issues in Engineering Education and Practice, ASCE, 168-178.

[106] Zack, J. G., Jr. (1999). "Pacing Delays, the Practical Effect", AACE International Transeations.

[107] Zack, J. G., Jr. (2001). "But-for schedules analysis and defense." Cost Engineering vol. 43/ no. 8, 13-17.