HiMoldeMaster

Master of Science in Logistics

Topics in Short-Term Planning at Ulstein Shipyard

Christian Nærø and Amar Mohite

Molde, 2008

Molde University College

Student Assignment for the Master Degree

Title: Topics in Short-Term Planning at Ulstein Shipyard

Author (-s): Christian Nærbø and Amar Mohite

Subject code: Log 950

ECTS credits: 30

Year: 2008

Supervisor: Åsmund Olstad and Halvard Arntzen

Agreement on electronic publication of master thesis
Author(s) have copyright to the thesis, including the exclusive right to publish the document (The Copyright Act §2).

All theses fulfilling the requirements will be registered in BIBSYS Brage, but will only be published (open access) with the approval of the author(s).

Theses with a confidentiality agreement will not be published.

I/we hereby give HiM the right to, free of charge, make the thesis available for publication on the Internet: ☒yes ☐no

Is there an agreement of confidentiality? ☐yes ☒no
(a supplementary confidentiality agreement must be filled in)

Can the thesis be published when the period of confidentiality is expired? ☒yes ☐no

Should the thesis be kept from public access? ☐yes ☒no
(according to the Freedom of Information Act §5a / The Public Administration Act §13)

Date: 28.05.2008

Preface

This paper is the Master Thesis of Christian Nærø and Amar Mohite, students at Molde University College, Norway. The research project is the final stage of our study programme titled Master of Science in Logistics.

The research topic has been executed under the guidance of supervisors Asmund Olstad and Halvard Arntzen. We would like to thank them for sharing with us their help and their academic expertise in their respective domains.

We are grateful to Ulstein Shipyard for giving us the opportunity to learn about practical life in the shipbuilding industry. During our meetings with planners and foremen we got useful insights about how things are conducted.

We would also like to convey our special thanks to Molde Research Institute, Research Assistant Karolis Dugnas and Chief Research Officer Oddmund Oterhals, who served as the links between us and the Lean Shipbuilding Project as well as the case company.

Christian Nærø and Amar Mohite

Contents

Appendix

List of Tables

List of Figures

1. Introduction and Research Objective

The Shipbuilding industry has been growing worldwide due to higher market demand for different types of vessels. Ulstein Shipyard encounters a similar situation regarding higher demand for offshore supply vessels but at the same time they have capacity problems concerning both infrastructure and workforce. “Lean Shipbuilding” and “Lean Construction” are evolving concepts which in part are adaptations of Lean Manufacturing into a project based production setting. Ulstein shipyard is currently in a process of adopting some Lean principles and tools in an attempt to deal with uncertainty and improve efficiency in production of vessels.

Our starting point in this thesis is that Ulstein Shipyard has low worker productivity. An exploration made by FAFO in cooperation with Ulstein Shipyard concluded that only 20% of worker’s time was spent doing actual value-adding work (refer to section 2.8). We also see that foremen at Ulstein Shipyard are doing short-term planning without conventional formal planning tools. Plans are not described in PERT/Gantt diagrams, and following this it is hard to make MRP calculations of different kinds. It is our hypothesis that **worker productivity is low because such formal planning is lacking**. We will not aim to prove or disprove this assumption, but we will suggest an alternative to their current planning methods.

Ulstein Shipyard uses the ‘Last Planner’ system (refer to section 3.3.2.3) in their short-term planning, and it utilizes a ‘Pull’ technique, not planning ahead but rather checking which activities are executable, making a pool from which activities are taken from to be scheduled. We perceive the reasoning behind this method of planning is that Ulstein Shipyard is dealing with a high degree of uncertainty internally as well as externally, and that deterministic methods do not handle such a high degree of uncertainty very well.

The objective of this thesis is to show that simulation can be used in conjunction with PERT/CPM, enabling these tools to handle a high degree of uncertainty more efficiently. A simulation model that can capture the stochastic nature of constructed sets of activities will therefore be built in Arena Simulation Software. This model will be used to demonstrate possible results of interplay between uncertainty, dependencies and durations of activities, giving alternative possible critical paths to which the deterministic approach PERT gives. We aim to present our findings from the simulation model and then give suggestions on how the information about these alternative possible critical paths can be used on issues regarding short-term planning together with PERT/CPM methods of trying to shorten the project span.

In Chapter 2, we will describe the shipbuilding industry briefly and give an overview of the focal company and its current efforts in planning. In Chapter 3, we present the scope of theories linked to the problems that we are dealing with. In Chapter 4, the simulation model is built. A detailed description about the model is presented including the data, experimentation with the data, verification and validation of the experiment. We then discuss the output obtained from simulation model and their interpretation in Chapter 5. The conclusion based on our findings and the research objective that we set will be provided in Chapter 6. Then finally, in Chapter 7, we imply limitations in our thesis and suggest further research.

2. Background

2.1. Industry Overview

Norway is one of the world’s leading suppliers of complex offshore-service vessels, and a cluster of companies designing and assembling such vessels is located in Møre and Romsdal. There are some challenges, however. The Norwegian shipbuilding industry in Møre and Romsdal managed to survive during late 70’s and mid 80’s by aligning it’s demand side towards offshore oil industry and shipbuilding for supplying offshore supplies as well as ship equipment for production purposes.

The industry claims they are continually experiencing challenges with competing internationally due to the high labour- and total production-costs in Norway. Due to the above reason, strategies of specializing on complex vessels were developed, because it was better for them to compete on cutting edge technology and solutions rather than cost alone.

The cluster in M&R is comprised of many companies in varying sizes and Ulstein is an example of a large shipbuilding company that specializes in advanced vessels; primarily anchor handling tug supply vessels, platform supply vessels and specialised and multifunctional vessels.

The supply market for parts and services is of limited size in the region, and we see that capacity constraints in the supplier market are something that the shipyards are struggling with at the moment. This gives external uncertainty when the suppliers are unable to complete an order because of capacity constraints. The uncertainty regarding deliveries is something one needs to take into account when planning production.

These are some of the reasons why some of the shipbuilders in the cluster have started collaborating with maritime research and development institutes as well as other shipbuilding companies, ship owners, suppliers, etc. in order to improve their performances. Ulstein Shipyard, Aker Yards and Kleven Maritime are involved in the Lean Shipbuilding project to improve their performances (refer to 2.4).

2.2. Company Overview

The Ulstein Group has in total around 600 employees and is established in Norway, Poland, Turkey, Slovakia, Brazil, China and Ukraine. The main yard, Ulstein Verft ("Ulstein shipyard"), has 380 employees and it is located in Ulsteinvik, in Møre & Romsdal county, Norway. It serves as the group's development base for assembly of capital- and labour-intensive, often "one-of-a-kind" projects: highly customized and specialized offshore/supply-vessels. Ulstein Shipyard is also the group's most important expertise base for carrying out projects, as well as a world leader on advanced vessels, mainly on offshore/supply vessels.

The shipyard's order books are in the first quarter 2008, filled-up for as long as 2011, something one could claim is common in the Norwegian shipbuilding industry at the time this is written. Ulstein has recently built state of the art supply vessels such as Bourbon Orca which have received international recognition mainly from the offshore oil and gas industry.

Ulstein Shipyard has capacity to produce only a few ships a year. Between September 2007 and October 2010 they are expecting to assemble and deliver 9 vessels (Sunnmørsposten, 2008). We see that their volume is low, so it can be assumed that it is a challenge to profit from the extensive R&D that is put into developing these heavily specialized vessels because serial production is hard to achieve with few vessels. It is argued that it is not the vessel itself that is earning money, it is the high tech equipment mounted on the vessel that has profitable margins.

Ulstein achieved a historically good result of 194 million Norwegian Kroner with a total turnover of 1.97 billion kroner in 2007 (Sunnmørsposten, 2008). This was after a very hectic year, where Ulstein spent a lot of time developing their final product and delivering 3 vessels. In a time where many shipyards are struggling with margins and late deliveries of vital equipment, this is quite an achievement. Late deliveries of equipment are something the shipyard has been dealing with continuously, but the shipyard has been able to deliver all the vessels at the agreed time in their contracts. This has been possible because of great flexibility in production, and quite probably also by having slack resources.

2.3. Physical Assembly of Vessels

It is common knowledge that the trend in the shipbuilding industry has been that they outsource the making of the work-intensive hull of the ship to shipyards located in countries such as Romania or the Baltic States. The reasons for this include shortages of qualified labour and labour cost. Ulstein Shipyard also follows this strategy; they outsource their hull-making to Baltic and Ukrainian shipyards.

After the vessel's hull is made, it is then towed to Ulstein Shipyard in Ulsteinvik, and they put it in the dry dock which has an area of over 100 meters long covered with walls and a roof to protect workers, equipment and the vessel from the weather.

The process of assembly of the vessels starts by assembling among other things, the superstructure of the vessel in sections. The sections consist of modules that are for the most part produced at their facility in Vanylven, south of Ulsteinvik. These modules are assembled into sections, outside the dry dock, exposed to the sometimes harsh weather. The modules are then taken into the dry dock and carefully lifted with a heavy lifting crane and mounted on the vessel. The limitation on how much this crane can lift is 250 tonnes, and this is said to be a bottleneck: if it could lift more, then more work could be done outside the dry dock. This is desirable, according to Ulstein representatives, because work-time in the dry dock (another bottleneck) is saved by taking as much work out of the dry dock as possible.

The work continues after the modules are mounted on the vessel. Workers are in this stage working inside the ship, in different areas including carpentry, electrical cables and equipment, painting, and so on. Many activities in the modules can not be carried out while they are exposed to the weather and moisture, such as insulation and electrical equipping.

Later, when the vessel is nearing completion of assembly, it is taken out of the dry dock and moored at a dock outside. It is then tested and launched upon completion.

Changes of design during the assembly phase are not uncommon at Ulstein Shipyard. Ulstein's customers often want the newest cutting-edge technology (which was developed after the contractual agreement) and these change-orders are uncertain and demand flexibility and learning in production. The possibility of such change is a strong advantage to Ulstein.

2.4. Participation in the Lean Shipbuilding Project

In order to maintain its competitive advantage, Ulstein Shipyard has been for a few years, and currently still is involved in a project named "Lean Shipbuilding" in cooperation with FAFO, Molde Research Institute and The Technical University of Denmark (TDU). The goal of project as a whole is to develop Lean Shipbuilding theory and also apply some of its lean practices to Ulstein.

It needs to be said that one should not blindly copy specific tools and techniques, because each organization and setting is unique and it is not always correct to assume that such tools and techniques are universal (Karolis, 2007).

Ulstein has realized that it is impossible to remove chaos, but you can learn how to live with it. The great complexity in and uncertain nature of shipbuilding has challenges that they want to overcome. Much of this is related to planning and management of the temporary organization of one-of-a-kind projects, which actually are uncertainties in it selves (Bertelsen 2000).

2.5. Planning at Ulstein

The Lean Shipbuilding Project lead to changes in how planning is done at the shipyard. In November 2006, the approach to planning named "Last Planner" was implemented for a vessel with the codename "build #277".

The planning of the vessel's assembly was divided into 2 main areas of responsibility: fore-ship/superstructure and aft/hull. This was done in order to simplify the planning process and we assume that the workers and foremen are somewhat specialized in their particular area and that the kinds of equipment to a certain degree vary between the different parts of the vessel.

The members of the Last Planner teams are foremen from different areas as well as external service suppliers and project managers, who sit together weekly and plan/re-plan which activities that should be carried in a 1-2 week horizon.

The motivation for not planning in detail long time ahead is strong in Ulstein. The need for flexibility due to the many uncertainties and risks a particular project is exposed to is an argument for this alone, and in addition the benefit from being able to postpone strategic decisions about the technical aspects of a vessel is a huge competitive advantage for Ulstein. Some customers want the latest technology, and they are willing to pay for it. The Ulstein representatives described this as "keeping solution space open". The figure below illustrates this:

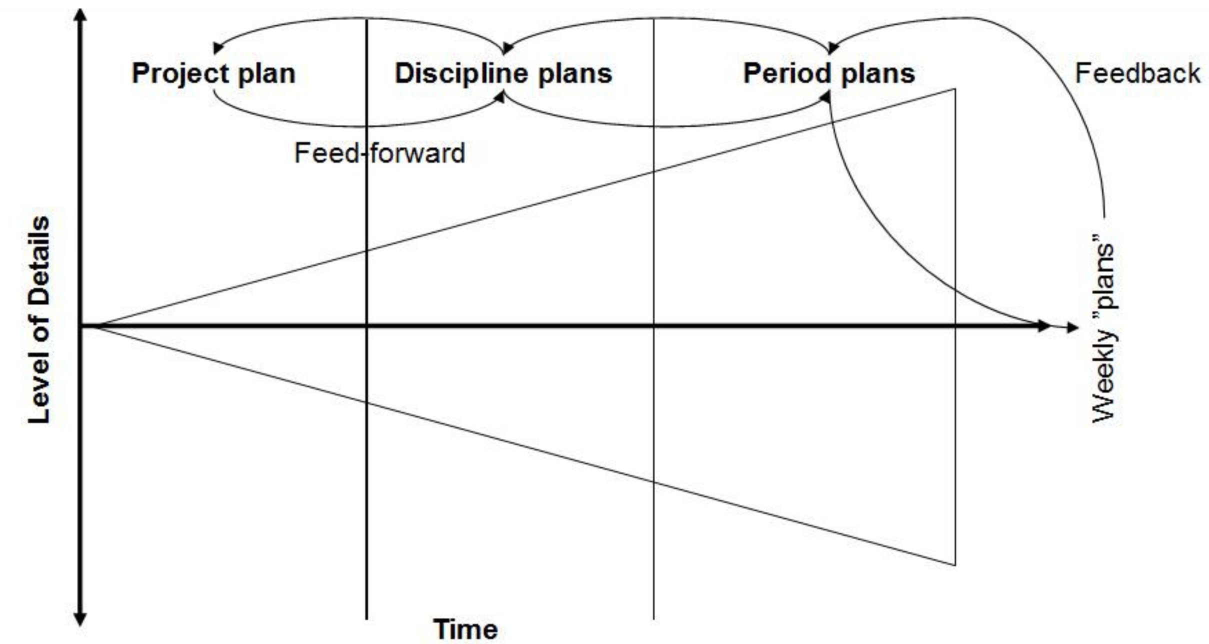


Figure 1: Detail in planning at Ulstein Shipyard

So far the implementation of Last Planner has according to the shipyard's representatives lead to a **higher degree of cooperation** between the foremen in different areas, reduced acceptance for delays of activities, the 2-zone division has made the planning more manageable and the degree of control as well as productivity has gone noticeably up. The week-plan meetings have helped charting uncertainties. The meetings also allow the foremen to systematically check that prerequisites for activities are fulfilled.

There are some issues, however. Still the shipyard is plagued with uncertainty. For example: Late arrivals of components are sometimes not known until 2-3 days after it occurs. This can be problem as components may be a prerequisite for a scheduled activity; the result is disruption of flow. The 2-zone division is also creating some potential headaches, there is a risk that the 2 zones may be performing isolated processes and as a result of this, competing for resources and causing suboptimal results.

2.6. Work Packages

The management and organization of activities is further broken down from the two main responsibility areas of the vessel. **Work packages** are defined as a set of jobs within the same area of the ship, requiring roughly the same worker skills and equipment and the progress and management of the individual work packages are supervised governed by foremen. We see that the shipyard might run the risk of sub optimization if resources can easily be reallocated within the same work package, but not so easily across work packages. In order to assemble a vessel, about 5000 work packages have to be carried out.

It is hard to accurately describe the uncertainty on the activity level, but we see that many activities can have uncertainty in:

- Normal duration: How long an activity should take? Historical data for similar activities can be used on forecasting this, but it will not be exact.
- Sequence: A planner does not necessarily know in which sequence activities optimally should be carried out.
- Interdependent relationships between activities in reality.
- What skills are needed to execute an activity successfully (ref: one-of-a-kind)
- What equipment or resources are needed where and at which point in time?

2.7. Uncertainty, Prerequisites and "Kitting"

The work associated with having all prerequisites ready for execution of plans is important and something the shipyard has been emphasizing a lot. The preferred jargon for this in the Lean Shipbuilding-project seems to be "kitting". We see that this concept is similar to Lean

Construction's 7 prerequisites (Koskela, 1999), which will be explained under the theory chapter "Lean Construction".

The figure below describes how different elements can be uncertainties that affect activities. A given activity can not start until the prerequisites are fulfilled. This includes internal factors such as having the eventual precedent activities complete in time, having and understanding the technical plans, having the material on site, having the correct tools available and having the people capable of doing the job. In addition to the internal factors, they also have to consider the external uncertainties that at are out of the shipyard's control: the weather being an obvious example of this, because some activities (such as welding of modules) are done outside in the open. It is hard to carry on work during harsh conditions that frankly are common in Ulsteinvik. Delayed arrivals of vital material/components to the shipyard are another potential problem that may delay scheduled activities.

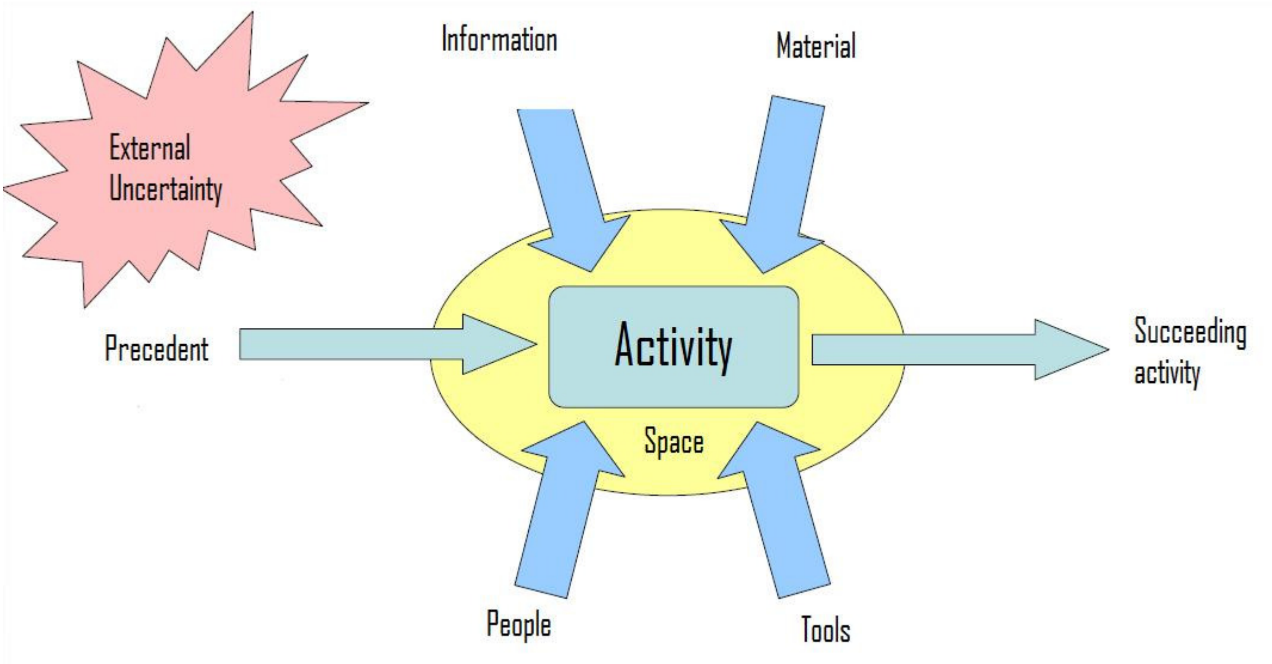


Figure 2: Uncertainties affecting activities at Ulstein Shipyard (Translated)

Planning of activities are not made any easier by the fact that it often is hard to determine how long a job normally should take, even under optimal conditions. Since the nature of projects often is "one-of-a-kind", sometimes the planners as well as the workers plan/do things that are new to them. Complexity to the picture is added when we assume that different workers are differently experienced, good at different things, and how well they perform can be dependant on who they work with, because of personal chemistry and complimentary skills.

2.8. Focus on Work Hours

Since assembling complex vessels is a very capital-intensive and skilled worker-intensive process, Ulstein has in cooperation with FAFO explored how time is spent by their workers. The results of this exploration are quite surprising, they concluded with that only 20% of a worker's time is spent doing actual value-adding work. The figure below describes their findings.

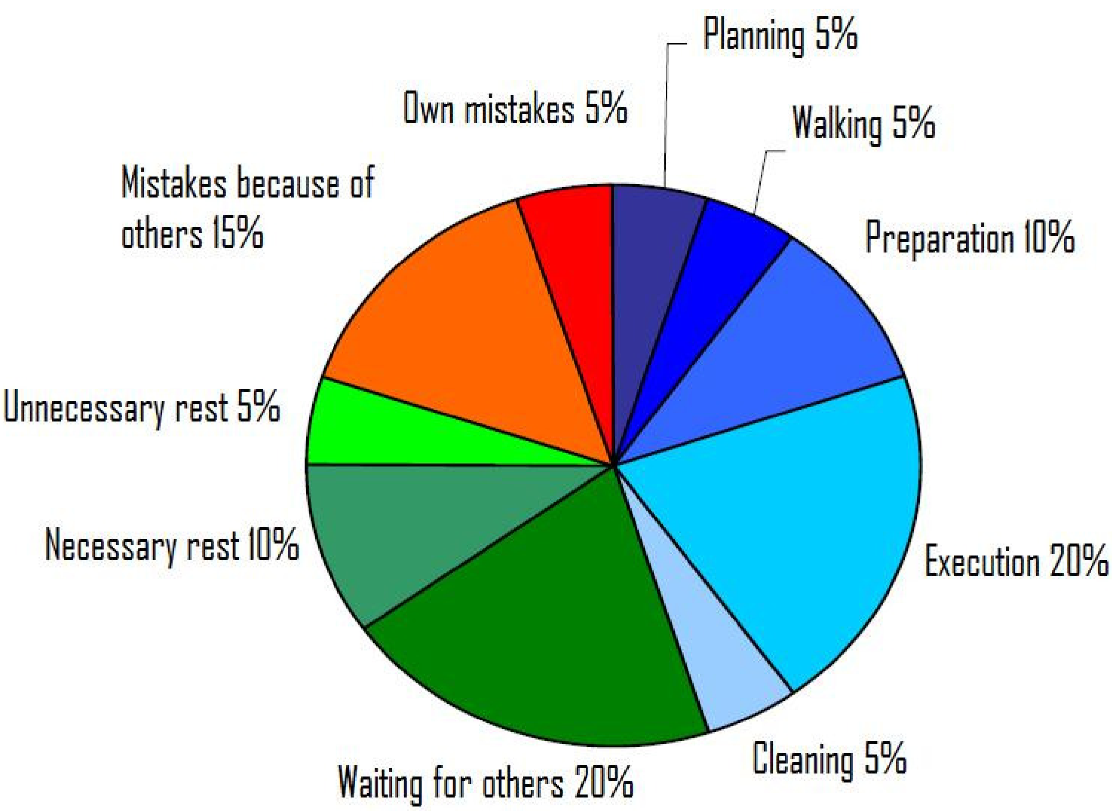


Figure 3: Distribution of total work time at Ulstein Shipyard (Translated)

The shipyard's representatives has made it clear that their ambition is that with improved planning, they can attack the "waiting for others"-part of the pie chart, which is measured to 20%. Having everything you need, where you need it/"kitting" can shorten the preparation and walking-time. These effects achieved through efficient planning can be substantial:

The advantages of improving worker productivity are obvious; not only because of the high labour costs in Norway, but also that if work is done more efficiently, especially in the dry-dock, they save time on the project and can assemble a higher number of vessels per year.

3. Theory Review

3.1. Project Management

A project can be defined as an array of activities or tasks undertaken to achieve certain objectives within certain internal factors like time limit, financial and human capital, as well as certain external factors which are difficult to evaluate and might only be forecasted.
Project management is a relatively new concept and demands massive precision due to the nature of customization of products in a competitive globalized world. Project management is attributed with methods of restructuring management for enabling better control and utilization of available resources thereby generating flow horizontally as well as vertically within a company.

According to (Kerzner, 2006:3), project management involves project planning and project monitoring at a macro level and defining of work requirements, quantity and quality of work per day/week, number of resources needed, evaluation and analysis of output and further actions or modifications on project parameters for better returns on the inputs at a micro level. In the same line of development, there can be numerous hindrances in project development process due to the uncertain nature of changes, complexities, technological changes, pricing issues, etc.

As decided, our project deals more with aspects of planning of activities, controlling and pre-emptive actions to mitigating uncertainties at the shipyard and in a complex environment where the cost of changes and excess of lead time might affect the competitiveness. In that case, we have to identify a particular activity area within the project in order to analyse and study its effects on the succeeding activities. It's equally important to identify the key initiatives for the success of the operating unit and to do so a methodology consisting of important aspects as the sources of analyses needs to be developed.

3.1.1. Work Breakdown Structure

"A project work breakdown structure (WBS) is a deliverable or product-oriented grouping of project work elements shown in graphical display to organize and subdivide the total work scope of a project"[1].

Work breakdown structure is a composition of organizational levels, technical levels, planning levels, execution levels and reporting/feedback levels. It enables integration of scope, schedule and cost during the life cycle of a product from point of origin to point of exit in the system.

Work packages are a term used for distribution of tasks at different levels and appraise the results on individual basis. The work package structure is also classified into short term work packages and long term work packages depending on the magnitude of the project. However, when work packages are smaller, possibly no assessment of work is needed and evaluation can be done at the end of work completion since objective indicators are easier to analyse with smaller work packages.

3.2. Program Review and Evaluation Technique and Critical Path Method

According to (Kazan, 2005:294), "Program Evaluation and Review Technique (PERT) is a network based technique for analysing a system in terms of activities and events that must be completed in a specified sequence to achieve a goal".

Each task in the PERT chart is connected to its successors (nodes) through links on the network. The network comes to completion only when all the nodes on it have been executed. It is suggested that project time can be estimated using both **Beta Distribution** as well as **Triangular Distribution**. PERT uses the probabilistic approach to calculate three different times namely the optimistic time, pessimistic time and the estimated time. It is likely that pessimistic times occur more often than optimistic for various reasons; one of which could be culture where one works according to a plan to meet the deadline but if something unforeseen happens (mistakes/pre-requisite unfulfilled), the deadline is exceeded. If the pre-requisites

[1] http://management.energy.gov/documents/WorkBreakdownStructure.pdf

arrive early, it may not have any effect on the project but if they arrive late, the entire project might be delayed.

This includes S*lack* which means that the duration for which a task to be processed is delayed without incurring delay to the consequent task and in the total project time. PERT facilitates with more dynamic approach to project management, control and planning of multiple activities and under which data for several repetitive projects can be generated and assessed (Page, 1989). PERT is a good technique mainly for scheduling, organizing and co-ordinating multiple inter-dependent activities and schedule based elements like resources, incoming material, etc. and also alterations in schedules periodically whenever uncertainty arises.

Project planning using PERT can be divided into several stages as follows:

- Deciding on the start date of the project
- Deciding on project completion date, both of these are good to know for statistical estimates of uncertainty.
- Generating project phases based on task dependencies
- Determining and assigning skill level and appropriate personnel to perform the tasks
- Project control methods can be executed by determining intermediate milestones set for particular dates

PERT is a model for effective project management which includes a terminology named critical path which is part of a scheduling algorithm called Critical Path Method.

The Critical Path Method is the collective approach of scheduling the project network which includes all the activities needed to complete the project, the duration each activity takes during the project and the dependencies activities have. Critical path takes into consideration the longest duration of project plan along with earliest and latest activities the project can start with, without making the project time longer.

“Du Pont developed CPM, which was particularly applied in the construction industry”[2]. The operational development of critical path started in parallel with PERT. There’s not much difference between CPM and PERT since PERT consists of CPM applications.

[2] http://www.referenceforbusiness.com/encyclopedia/Per-Pro/Program-Evaluation-and-Review-Technique-PERT.html

The main idea is to improve the more comprehensive processes that fall on the critical path by assigning relative resource and scheduling priorities to them. All the other processes that are incidental can have substantial flexibility in requirements as they can be assumed to be finishing before the critical ones. But, according to (Puich, 2007:28), “All contributors to the process should have clear understanding of their capacity and see their work activities as a priority, regardless of where they fall on the critical path”.

It can be perceived that control over all the processes is must in case of changeover on the critical path. In such cases, the non- critical activities must get aware that they are on the critical path due to delays caused by lacking internal operational performance or uncertainties covering the project.

When a project encounters complexities and variability with processes with rigid times, proper review and supervision is needed to ensure that all the elements on those processes are aware of their completion dates. This is tact under project management to assure that the work is in progress through setting up of intermediate milestones which also gives picture about errors, rework time and also prompts over re-planning approaches to ensure position of the project on the right path.

An exemplary critical path diagram is shown below with the red rectangles being the critical path. There are four sub- rectangles with the top three mentioning early start time, duration and early finish time; the middle one is for task name and the bottom three are for late start time, slack and late finish time.

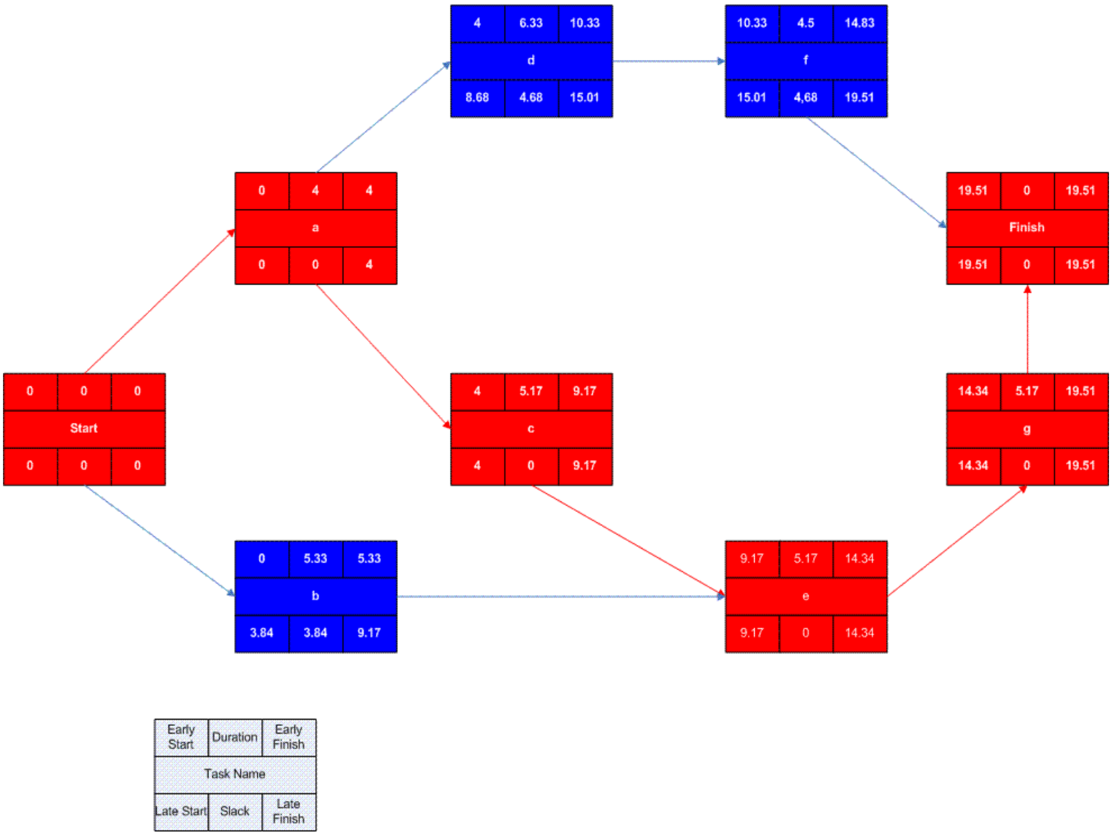


Figure 4: Exemplary Critical Path Layout[3]

3.2.1. Steps in Critical Path Method

The execution of CPM is guided by certain steps which is necessary for determining the critical and non-critical activities along with their durations.

Step 1: Listing the Activities

The activities are listed in a table along with their predecessors and how long it takes to finish that activity

Step 2: Drawing a Network

After identifying the activities in the project, a layout showing nodes and arcs representing stages of project completion and actibity number/name, respectively. The activities with no predecessor get on the first node and the activities with no successor get on the highest node.

[3] *Wikipedia, Program Evaluation and Review Technique, at* http://en.wikipedia.org/wiki/Program_Evaluation_and_Review_Technique (accessed 19 May 2008)

Step 3: Back-Tracking of Critical Path

The critical path of the project can be back-tracked using activity durations. We start with the last finished activity, take its start time and look for the same start time in the finish time column for the activities above. The activity that has the same finish time as the start time of another activity, then that activity is the predecessor to the activity with same start time.

Step 4: Calculation of Crash Cost/Time

We assume that time can be interpreted in terms of money, as more work done per unit of time normally means higher profits. After determining the critical path, crash costs for activities on the critical path are compared against how worthy/time saving it is to crash that particular activity. We also assume that crashing means to reallocate resources from non-critical to critical activities in order to shorten the duration of the project as a whole.

3.2.2. Advantages of Critical Path Method

- Critical path is initiated with total work breakdown of the project and help in estimating the manpower and the kind of skills required by them to take on the tasks on the critical path. Critical path also helps in sketching map for precedent processes thus completing the chain.
- Identification of path in terms of work hours helps in planning of workers with an intention of shortening the aggregated time needed. Proper estimation and availability of resources is needed since it is difficult to mitigate time extension if delays occur on large scale projects. Especially in construction industry, it is not easy to find appropriately skilled work force in a short time if problems occur in the middle of the project. CPM also facilitates notification of shortage of work force well in advance to match the plan with reality.
- Superiors can rely on some base to build a final plan and move onto to scheduling practices for the project until they find a better tool to plan accordingly and improve the output and lower cost.
- CPM deals with uncertainty to a large extent. Projects with high capital involved run into many uncertainties during short-term as well as long term planning and it is must to be invulnerable (if not entirely) to such uncertainties since all the supply chain links

to that project become tight which might cost more if uncertainty is not mitigated through better planning.

Gantt charts are used to show relationships between dependencies and predecessors using a time scale as to when the activities and the project will be completed. Gantt charts represent the work breakdown structure of the tasks comprised by the project and the durations for those tasks. An exemplary Gantt chart with tasks, dependencies and durations is shown below:

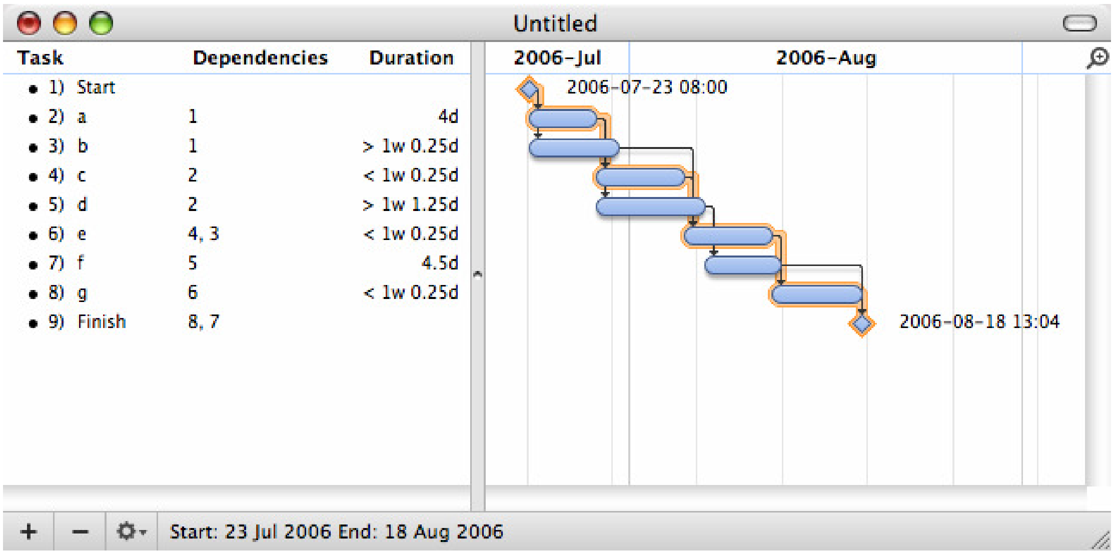


Figure 5: Exemplary Gantt chart[4]

3.2.3. Crashing and Time-Cost Models

"Crashing is the method of reducing project completion time by bringing in additional resources on an activity along the critical path of the network", (Haga, 2001:1). The method of crashing has been practiced for many years on small as well as big projects but still many projects end up with delays or excessive project costs after completion. It's is said that projects with an obvious critical path have at least 50% probability of finishing that particular network on due date but on the contrary projects with multiple critical paths have much less than 50% probability of finishing the network on due date. The reason for this is the complexity over allocation of resources to 'which' critical path over the network.

[4] *Wikipedia, Program Evaluation and Review Technique*, at
http://en.wikipedia.org/wiki/Program_Evaluation_and_Review_Technique (*accessed 19 May 2008)*

Thus, such project networks are interpreted using Time-Cost models where crash cost and overrun cost are assigned to activities crashed on the critical paths. Time-cost models are significant in determining which activities on the critical paths reduce the total crash cost plus overrun cost when selected for crashing and that crashing is continued until the total crash plus overrun cost continue to show decline. The base comparable parameters for this are the mean completion time and the mean crash plus overrun cost.

3.3. Lean Theory

In this section we will define Lean Manufacturing, Lean Construction and Lean Shipbuilding. The lean theories and concepts are described because they are needed to understand the reasoning behind the planning methods being applied at Ulstein Shipyard.

3.3.1. Lean Manufacturing

Lean Manufacturing is essentially a philosophy that seeks to eliminate resource wasting in a supply chain setting (Harrison and Hoek, 2005). Lean Manufacturing is today practiced in the Toyota Production System (TPS), first thought up by Taiichi Ohno at the Toyota Company during 1952-1962 (Liker and Lamb, 2000). Toyota remains a pioneer in Lean Manufacturing and is today very famous for lean tools such as "Just-in-time", although they claim that people, culture and learning has been the real foundation of their success (Liker, 2004).

TPS was originally inspired by both the benefits and the problems following Henry Ford's notion of mass production: standardization of processes and creating continuous material flow but also allowing wasteful batch production that built up large volumes of work-in-progress inventory in a push-oriented way of production. TPS's critique of mass production is that excess inventory is an important source of waste and that it can potentially hide quality issues as well as triggering costs associated with handling, storage and capital.

Toyota defined '**muda**', which means 'waste' in Japanese, and the term, is used in general to describe wasteful activities or activities that doesn't help add 'customer perceived value' to the end product. We see that there are two types of waste: **necessary waste**, which one should seek to minimize; and **unnecessary waste**, which one should seek to totally eliminate.

The concept of 'the seven manufacturing wastes (plus one)' followed this and they are defined as (Liker 2004:28-29; McBride, 2003):

1. *Overproduction* – Prohibits smooth flow.
2. *Waiting* – Downtime in a bottleneck affects the output of the entire chain.
3. *Transporting* – Costly non-value adder. Transport can lead to damages on goods.
4. *Inappropriate Processing* –Work can often be done with cheaper equipment.
5. *Unnecessary Inventory* – Is a cost in itself, but it also hides flow problems.
6. *Unnecessary or excess motion* – Ergonomic issues.
7. *Defects* – Defects can disrupt flow and rework and scrap can be very costly.
8. *Underutilization of Employees* – One should capitalize on employees' creativity.

The concept of reduction of waste in production is heavily emphasized in Lean Manufacturing, because when waste occurs, we assume that production is happening under sub-optimal conditions. One might argue that Lean Manufacturing is a system under which some common sense is organized ("waste is bad").

Several 'lean' guidelines for production have been proposed (Liker, 2004):

- *Use 'pull' systems to avoid overproduction ('Just in time')*
- *Level out the workload (smooth flow)*
- *Build a culture of stopping to fix problems, to get quality right the first time.*
- *Standardized tasks and processes are the foundation for continuous improvement and employee empowerment.*
- *Use visual control so no problems are hidden.*
- *Use only reliable, thoroughly tested technology that serves your people and processes.*

Lean tools such as the above have been widely adopted by production companies throughout the world following Toyota's success as a lean car manufacturer, but according to Liker's maybe controversial statement in 'The Toyota Way' (Liker, 2004): most 'lean manufacturers' are only scratching the surface of true leanness.

We see that different manufacturing companies for the most part produce very different kinds of goods, but they all face similar problems when it comes to wasteful activities. This is alone a strong argument for adopting leanness. It should however be noted that historically, Lean

Manufacturing has most commonly applied to a situation where product flows through a production line, as value is added to it.

3.3.2. Lean Construction

We see that Lean Construction is a relatively new concept with some amounts of literature available, and it is clear to us that Lean Construction is an adaptation and translation of lean thinking to a situation where the 'production' is one-of-a-kind project based, and situations where equipment, material and personnel are flowing around the product as opposed to lean manufacturing which we previously described. In essence, Lean Construction aims to emphasize reliable and speedy delivery of value, and to challenge the belief that there is always a trade between time, cost and quality[5].

3.3.2.1. Lean Construction Institute

The Lean Construction Institute was founded in August 1997[6] as a non-profit organization and has since then worked to reform the management of production in design, engineering and construction for capital facilities. They claim on their website (view references) that they have produced significant improvements, particularly on complex, uncertain and quick projects. The planning system 'Last Planner' was invented by Lean Construction Institute. The International Group for Lean Construction[7] is also an organization working with this, and they have since 1993 published papers on the topic.

3.3.2.2. Principles of Lean Construction

Finnish Professor Lauri Koskela, a major contributor in Lean Construction literature, has given 14 principles of Lean Construction (Dugnas and Uthaug, 2007):

'*1. Reduce the share of non-value adding activities (waste).*

2. Increase output value through systematic consideration of customer requirements.

---

[5] The Lean Construction Institute: 4th Annual Lean Project Congress 2002, at http://www.leanconstruction.org/pdf/LCIWebBro9.pdf (accessed 20 May 2008)
[6] The Lean Construction Institute, at http://www.leanconstruction.org/ (accessed 5 April 2008)
[7] The International Group for Lean Construction, at http://www.iglc.net/ (accessed 5 April 2008)

3. Reduce process variability. Consider process interdependency and isolate supply-related variation.
4. Reduce cycle times. Eliminate inventory stock and decentralize the organizational hierarchy.
5. Simplify by minimizing the number of steps, parts and linkages in a product and the number of steps in a material or information flow.
6. Increase output flexibility. Use modularized product designs, reduce the difficulty of setups and changeovers and train a multi-skilled workforce
7. Increase process transparency.
8. Focus control on the complete process. Allow autonomous teams to exercise control over the process and build long term co-operation with suppliers.
9. Incorporate the best practices into the organization and combine existing strengths with the best external practices.
10. Build continuous improvement into the process.
11. Balance flow improvement with conversion improvement.
12. By improving performance at the planning level increase performance at the project level. The Last Planner method is an appropriate alternative.
13. Shift the design work along the supply chain to reduce the variation and match the work content.
14. Benchmark'

3.3.2.3. Planning in Lean Construction

Last Planner is a production management system developed by Lean Construction Institute.

The levels of planning are divided into four:

- Project plan that charts the rough outline a project, 15-18 months (Master schedule).
- Long term discipline plan that estimates what should be done and when, 6-9 months (Master schedule).
- Medium-term period planning that prepared for execution of activities with 6-8 week horizons. Also called the look-ahead plan.
- Short term planning in a 1-2 week horizon, picking from a pool of executable activities, which will be carried out according to the plan.

Plans for production begin with a master schedule; the project leaders would have their main focus on the long-term process plan and the period plan. This puts the focus for them on a 6 week and upward horizon. Plans for six weeks ahead are evaluated and analysed by them. The goal is to find out if any constraints can hinder the success of the production process. Koskela suggested 7 general prerequisites in Lean Construction shown below in Figure 6 (Koskela, 1999):

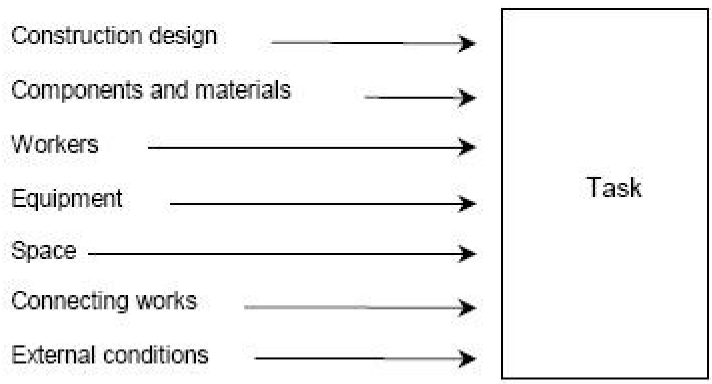


Figure 6: The preconditions for a construction task (Koskela, 1999)

After the master schedule and look-ahead plans are established, the ‘Last Planner team’ come together weekly, discuss the previous and coming week and make, given that the resources needed are available (constraint analysis), written commitments about the following 1-2 weeks.

The look-ahead plan is central in making these short-term plans, because it actually charts executable activities and then the ‘Last Planner team’ on a weekly basis discuss and then chooses from these **‘executable activities’**. In doing this, a ‘pull’ oriented scheduling is created, decoupled from the ‘push’ scheduling in the master schedule on the higher levels of planning. The philosophy behind this ‘pull’ scheduling is that it gives flexibility and flow of work.

The scheduled activities that are actually carried out (success) or not (failure) should be recorded and compared with the commitments made in the previous meeting, so they can learn from their mistakes by for example making statistics on why activities were delayed, et cetera. We see that the Last Planner is a closed loop control system, and that conventional ‘push’ (MRP) plans are known to produce large swings in plans when adjusting for small variations in production, making them relatively useless for detailed shop floor planning. In Last Planner, the plans are made by real people who can adjust according to weather, competence of workers available, resources available, or other constraints. The reliability and flexibility in such plans has proven to give productivity gains (Poppendieck LLC).

The involvement in planning can is also possible to be extended to external suppliers of services that because of this can better plan their capacity when they are allowed to participate in the planning process and receive better information about what is required of them. We see that this can contribute to reduce external uncertainty.

3.3.3. Lean Shipbuilding

“Lean Shipbuilding” is an emerging concept, closely related to “Lean Construction”, which is a recently developed concept, aiming to apply lean thinking to project-based production. These concepts are in turn derived from lean manufacturing.

We see that the original concept of lean was proposed for mass production where value is added while the product flows along a production line, not project-based one-of-a-kind / engineer-to-order (ETO) production, like in shipyards, where the product, the vessel, is static and the workforce and equipment moves around it. Obviously there is a great difference between these two ways of production (mass production vs. project/workshop) and lean techniques must therefore be differently implemented.

The basic principles in lean of satisfying customers with shortened lead times and enhanced value and quality, by eliminating waste in any process should be very interesting for any shipyard.

It is worth noticing that Norwegian shipbuilding is quite different from American or Japanese practices (Liker and Lamb, 2000). While they can gain benefits in standardizing the vessels, Norwegians, however, are often extremists in customization where the commonly used term ‘one-of-a-kind’ is a literal description of reality. Following this we see that standardization of some components or processes can be problematic at Ulstein Shipyard, but it must be kept in mind that processes can be set up in such a way that they can be beneficial for different projects. Establishing routines in planning, reporting, learning, error prevention and failure analysis can be such common denominators.

3.4. Theory of Constraints

Theory of Constraints is a management philosophy developed by Eliyahu Goldratt, an Israeli physicist, since the 1980s. Goldratt wrote several business novels, including "The Goal" (1984), "It's not luck" (1994), "Critical Chain" (1997) and "Necessary But Not Sufficient" (2000), as well as books on Theory of Constrains: "The Race" (1986), "Theory of Constraints" (1999) and "The Haystack Syndrome" (1991).

The philosophy conveyed in his books is concerned with performance improvement in production. Goldratt's idea about production is that "an hour lost in a bottleneck is an hour lost in the entire system" (Goldratt and Cox, 1984: 158). Following this, strong focus on dealing with bottlenecks is natural in the philosophy.

Goldratt developed five principle steps of Theory of Constraints (Goldratt, 1997):
1. *Identify the (primary) constraint (bottleneck).* There are four types of bottlenecks: physical constraints, supply chain constraints, market constraints and policy constraints; where the latter is often the most important, as it describes how things are done within the organization.

2. *Find out how to exploit the constraint as much as possible.* That means that we should try and maximise the output of the constraint, without investing in additional capacity.

3. *Subordinate everything else to the decisions made in step 2.* This means that we should acknowledge that the bottleneck determines the systems output, and is therefore most important. If resources lack in the bottleneck to keep it from operating at near 100%, then borrow from other non-bottleneck activities.

4. If *it is still a constraint, elevate the constraint so that a higher performance can be achieved.* We invest (money) in adding capacity to the bottleneck. Make sure you are investing in the right place.

5. *If the constraint is eliminated, go back to step 1.* This will force you to rethink your system all over again, as you will have other bottlenecks now. Using these 5 steps is a continuous process, and because of this it should give continuous improvement.

Theory of Constraints has served as an inspiration in development of Lean Shipbuilding (Bertelsen, 2008: 8) because when bottlenecks are dealt with, flow is improved. **We see that a bottleneck in a project-based context can be a crucial activity other scheduled activities are dependant upon,** as well as the original idea of bottlenecks in resources (machines, workers, et cetera)**.** This reasoning is shared with Critical Path Method which states that it is the 'critical path' that dictates the duration of the project (explained later in this thesis). The point is that if the bottlenecks (critical activities) in a system are known, one may be able to improve the performance as a whole by focusing on the bottleneck.

3.5. Simulation

Simulation is a technique of emulating practical activities/operations using systematic and specialized tools. In present days, computer simulation has become an integrated part of modern technology and its implementations.

Computer simulation as we know consists of a model facilitating a conceptual system. In our case, the system is the shipyard and the different sectional activities at it. The system can be anything right from a restaurant, railway ticket booking counter (which can help to forecast demand and align capacity accordingly), airline online ticket cancellation system, car arrivals at parking lots, etc.

The primary reason for using simulation as a tool is because with simulation models we can be able to take into account the stochastic nature of the project network which might have multiple critical paths on it. As opposed to PERT, which ignores the stochastic nature of activity times and which lacks in defining crashing of project network with multiple critical paths.

Simulation can also be helpful in doing comparative analysis of what has happened previously and what will happen if certain steps are taken in a particular direction. For instance, if we have to compare the performances (mean time/cost and expected time/cost) then simulation is a good tool. In reality, applying trial and error method could be too devious and too costly. If the shipyard goes onto adopting a new resource allocation and schedule pattern instantly, there will be delays due to bottlenecks, improper schedules, shortages of resources, etc, which will emerge as a blunder. However, our efforts will be in the direction of building a model which will enable us to logically conclude using comparable results. The model will be interactive and logical representation of a system. A logical computer simulation model is something in which the logic behind making the simulated model is to be understood and reasoned. This is possible only by protracting the real situation around and within the system and then mimicking it into virtual model using information accumulated from real world and using mathematical tools and software.

We will experiment the model to help formulate a more effective plan under uncertainty. We will show how different degrees of uncertainty and what impact uncertainty has on possible critical paths since simulation can give us a feasible picture of what can happen even before a project is implemented and executed.

We also plan to play with the system as much as we can within the time constraint since it will help us see how the model behaves and how the output changes under different uncertain situations.

3.5.1. Validation and Analysis

While we build our logical model, we should ensure that our logical model replicates the real situation which is referred to as model validity. Validation can be easy with small sized problems which deal with so-obvious situations. But, when the system becomes complicated, validation might prove to be very time consuming and difficult.

Animation might prove to be useful during circumstances for verification and validation levels as it allows observing the entire system in full mode and the behaviour can easily be tracked as the model performs. Other ways are statistical tests or comparative analysis for further edge into the problem. Such checking methods will help us perceive the system behaviour better and more can be seen through if we can perceive possible situations with longer duration for model run which will give us a range of result and thereby make us think within that range. We can try playing with the model with discrete set of ideas and look at and analyse how the model functions under every discrete set using numerical evaluation which might not be possible to do in a real situation.

3.5.2. Benefits of Simulation

As we have discussed before, simulation is an analytical representation of an experiment that we do with our system. And thereby, we intend to highlight on how this experiment aids us with a combination of pre-emptive visualization and post analysis that aid in making better plans.

Firstly, simulation helps in storage, execution and display of data and results that can easily be forgotten by us. The massive and complex nature of the problem can be dealt with effectively using simulation, as opposed to the human mind where our ability to do take into account more than a few variables is limited.

Secondly, simulation can also act as a mode of sharing information and understanding the system. Through visual display of the system, the way the system behaves (subject to changes in parameters), can be explained to anyone quite more effectively than putting and interpreting the matter on paper. This is also one of our major objectives for this thesis that we will show the interplay between uncertainty, durations and dependencies in constructed sets of activities, similar to the work packages activities are managed within at Ulstein.

Thirdly, due to dynamic nature of simulation, it helps us to foresee and predict the challenges and shortcomings in the future. Information and data available with us from past and present can be used to speculate happenings in the system in the future. For instance, if the plan today is meant to be executed for 6 weeks and see the results, with limited options in the available simulation software we can actually run the model for as long as we want to or as long as the project time to see how it behaves in a long run with real parameters and how the target deviates from the one benchmarked upon. In other words, it is like experimenting over the system until satisfactory results are achieved. It’s more effective and cheaper to experiment with an unreal system than a real one which in turn doesn’t result as an impediment either.

In fact, there are a lot more benefits of using simulation in general, but due to the nature of our project and time-constraint, we have to limit our line of thinking based on the points given above. For instance, we are not dealing with resources or optimization in our thesis.

3.5.3. Difference between Simulation and Optimization

Simulation is more like a statistical tool which gives a range for results like an average plus/minus the confidence interval. But in optimization, we can get an exact/optimal value for the input data or a value that is very close to the optimal. Simulation gives us estimates but those estimated can be relied on in terms of percentages based on replicating the system several times which can probably say that the result is authentic certain percent of times. According to (Bratley, Fox and Schrage, 1983), "Your costs will be A if you take action X,' but it does not claim anything like, 'Cost is minimized if you take action Y".

The benefits of simulation and optimization are somewhat similar which enable better quality analysis, better ability to understand the effect of dynamic events, better understanding of the system and most important cost savings and risk mitigation.

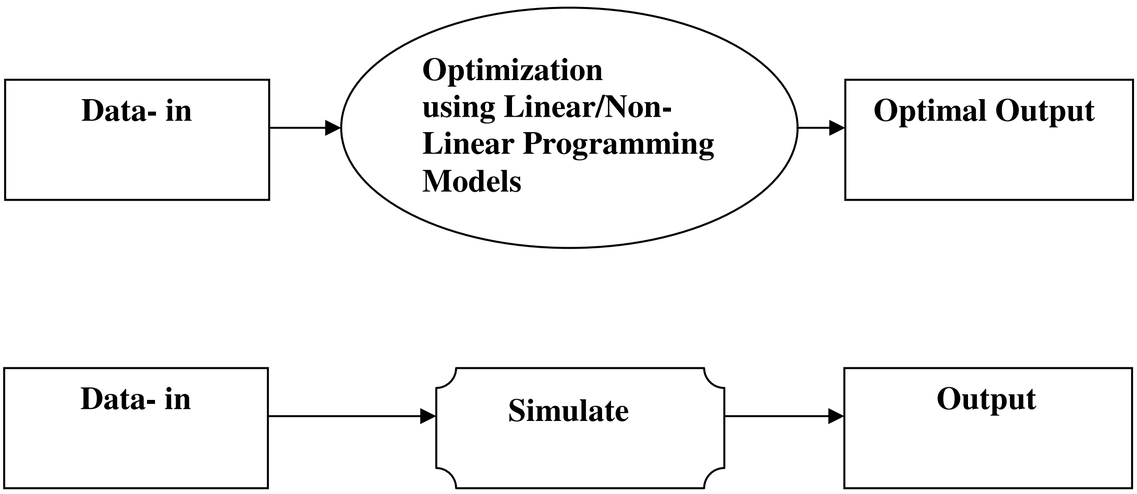


Figure 7: Format of Optimization and Simulation

Simulation allows us to see the behaviour of the system over time as the inputs change. It allows us to identify the bottlenecks that may not be possible to do so when the optimization interval is running. But to find the optimal answer we have to observe what has occurred then make some hypotheses about the nature of the system and make changes in the attributes, for instance number of resources/ work hours/ inventory levels, etc. and see what happens. Optimization and simulation is possible to use together as it is rapidly becoming common in industrial areas within the supply chain. The main motive behind simulation is analysis and not optimal result. The output can be used to explore the area broadly and try to improve the system accordingly.

3.5.4. Testing Issues with Simulation Model

A simulation model can be of various types ranging from service models (Distribution, hospitality, banks, etc.) to manufacturing (Industrial automation, Warehouse management, etc.). Our model will be part of the manufacturing aspect which will deal with activities under work packages to be performed within a specified time. The ability to build a model will give us insights into the planning aspects as well as how the plan can be altered to improve the operations at the shipyard. We'll also be working on drawing the critical path for the cases we'll be considering. As we know that critical part is a vital part of project management and also that critical path is vital in recognizing the most important, most time consuming and most value adding activities/task. In our case, we will use time required for the activities as a base for analysing critical path.

Our testing will also provide reasoning and argument about how the estimated output from simulation differs from that of PERT (Program Evaluation and Review Technique, the results will be shown in tabular form with Gantt charts) but there are some issues while building a model that need to be addressed.

Our model has a setting based on a simple data provided to us regarding work hours for work packages. We also need some specific comparative statistical data generated through PERT to see whether our model responds positively to our needs.

The model is based on the concept of generating dependencies from precedents which in our case are activities whose dependencies are denoted with a matrix. For this reason, we need logic behind delaying the activities until the matrix is screened through by the dummy entity. So, we needed a logical variable to define this matrix. During the model run, you will see a queue at one of the modules defined for holding the dependent activities and start activities, the basis of whose logic is very important for respecting the activity initiation structure generated through PERT.

It is very important that we understand how the data available has to be used on the model in order to make it function. The logic behind it is to read the data using a sub model before the activities are generated and held at the module. We no longer need to use the variable matrix in Arena but have to create a logical variable matrix using the sub model which transforms the data from Excel into Arena and aligns the dependencies with single dimensional figures for optimistic, normal and pessimistic times.

The proper collection of this data by Arena will influence the performance of the system and will be vital in developing further logic for generating critical path using Arena.

3.5.5. Explanatory Principles

As discussed before our explanations and arguments will be based on the critical paths achieved which intended to influence the make span of the entire set of work load during a specified time which in turn will help influence the dynamics of scheduling and bring efficiency. Make span is the difference between the end time and start time of a particular activity or total operation. We base our ranking of activities on the basis of general perception due to inadequate information about their sequence. So, our priority will be based on the perception about activities with longer processing times, more dependencies and also inflexible resource usage with a resource with capacity equals one will be capable of handling each activity individually.

3.5.5.1 Processing One Activity at a Time

The ships that are being built at Ulstein are basically supply vessels to the offshore industry which are as large as 30*110 metres (height and length), approximately. So, they comprise of superstructures and also medium and mini structures which can weigh as much as 250 tonnes, each. Making these structures is time consuming based on definite plan. From the above information, it is clear that every activity under the heading work package (could be any sized structure) utilizes a limited resource which in this case is space with limited capacity of 1 assigned in Arena. With random nature of the model, the activity will be assigned to the space (resource) that is vacant in numerical pattern. The logic also suggests that all activities should be covered in the manner as they are represented in the logical variable matrix. This will ensure no salvage or scrapping of activities if the resource is unavailable. Due to dependencies and sequential usage of resources we might not require more areas (space) to conduct work on since having more of them idle

4.5.5.2 Activities with Dependencies and Longer Processing Times

We will assume that 2 or 3 activities start at the same time but have different end times with a minor difference amongst them. The reason for doing so is that it will give us insights into how the critical path might vary with the use of simulation under circumstances of uncertainty which will be generated by probability distribution.

It can also be the case that we get different critical paths for different replications that we run the system for.

There is difference between service system and manufacturing system (in our case) and in that there is possibility of balancing the utilization of capacity depending on the alternatives available. For instance, in service system, preferences can be given to services and even Arena can aid in doing that with the command called first- in- First-out or in other words First-come- First-serve. And also if the resource capacity does allow, the service can be shifted to the next resource with under- utilized capacity.

But in our case, the dependencies cannot be released unless and until the precedent is finished irrespective of available capacity or little work to be conducted on the dependency because the dependency can be a minor part of the precedent. So, in other words it is quite difficult and tedious to balance the resource capacity unless the tentative time required on critical path is known.

4. Simulation Model

As we have described before, a simulation model is the replica of what happens in the real world. Simulation consists of three main components namely the model inputs, model execution and analysis of the output (Fishwick, 1994). It will also comprise of model testing, analysis of experiment results and validation. After the model has been presented, its parameters (data), logic based on assumption and constraints will also be described in detail.

4.1. Model Inputs

4.1.1. Activities in the Model

Three separate cases with same 33 activities but different optimistic and pessimistic time assumption over the normal times will be taken. There will be three activity branches running parallel to each other and preceding the final stage of the process. After finalizing the dependency structure, a Gantt chart will be constructed to visualize the structure better.

4.1.2. Work Hours

Data regarding work hours for activities has been self-constructed by us. Taking at as a base, we calculated the optimistic, pessimistic and estimate time {(optimistic + normal + pessimistic)/3} using **Triangular Distribution**. Optimistic times are those that show improvement in the performance of the processes. Pessimistic times are those that reveal negativity in performance caused due to delays on processes or other internal and external factors. Normal time is the mean time that the processes/activity will run for. All these things are calculated for each activity and for all the three cases.

4.1.3. Dependencies

Dependencies will be assigned using a logical variable matrix. The data will be input and extracted from an Excel spreadsheet at the same time as the work hours will be extracted. The Gantt chart showing precedents and dependencies will also be presented with the cases. The system will use the logical variable matrix to identify the dependencies and will also use the corresponding activity work hours with **Triangular Distribution** during runs.

4.2. Execution of Simulation Model

The model has been using Arena 9.0 over Windows XP. The model is flexible to modifications in cases that we are considering. The data will be extracted from Excel and the primary output (mainly critical path) we are seeking for will be delivered to Excel itself in addition to Arena results that will provide us with queuing, utilization, waiting time, etc.

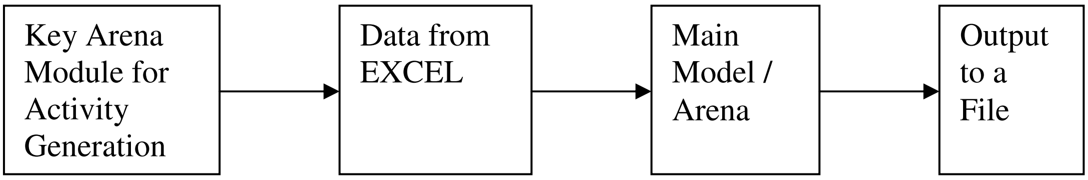


Figure 8: Layout of Combination of Data Flow and Logical Model

4.2.1. Key Aspects of the Model

Every virtual or technical thing has some commands, keys or language that is understood by that particular programme. In Arena also, we have to keep in mind certain technical aspects while building and executing the model. These aspects start from the beginning, become very complex and imperative in the middle of model building and relax when the end approaches since the key part is dealt with in the middle.

All models have common aspects which are Entities, Attributes, Processes and Variables which actually define the model. In addition, we will also explain different sub- models constituting the most important part of model functioning. These sub-models are vital in generating, releasing and checking dependencies and corresponding work hours for activities.

4.2.1.1. Basic Aspects of Arena

Entities, Attributes, Processes and Variables are the basic components of every model. Without these, it is not possible to run a simulation model. An entity is vital in initiating the system. Entities upgrade the actions on the process path which is known as event calendar. In order to do so, entities pass through different modules that imply different actions to be conducted such as holding the activities, separating, etc.

A *Create* module is used to create such an entity in Arena. The entities upgrade the process path through connectors that connect one module to one or more. Some minor but key adjustments can be made in this *Create* module like arrival rate of entities as well as time between arrivals supported by distribution and even a specific schedule. At the end, all the entities that pass through the entire system are disposed using a *Dispose* module.

4.2.1.1.1 Entities

As discussed above quite thoroughly, we have understood what role entity plays in the model. The entity is created in *Create* module under the heading *Entity Type*. Under *Entity Type* the name of that entity is assigned to remember the nature and role of that entity. In our model, there will be two entities namely a dummy entity called *Start Entity* and *Activity*. The two *Create* modules holding these entities will be placed parallel and the *Start Entity* will trigger *Activity* entity since the dummy entity is used to check the logic placed further in the path of system and then start creating the real entities if the logic applies to them.

The entities will be screened sequentially by the logic used and then a queue of real entities will be developed at one of the modules used for holding these *Activity* entities as the system logic is performed on these accepted entities.

4.2.1.1.2 Attributes

In computing attribute is known for signifying the property of an element in emphasis. For instance, in simple words there's a cloth which can be an element in computing and the colours that cloth is available in are the attributes to that element (cloth). All entities might have the same attributes but the value of those attributes need not be the same.

Thereby, attributes are entity dependent. In our model, entity dependent attribute is the *Activity Number*. So, every entity has a number but the numbers are different in the same way as every entity (*Activity*) has work hours but they differ. The Attributes can easily be adjusted and checked by using the list appearing below the model space. The information under this category is according to our test model. For the later part of analysis, relevant changes will be highlighted for better understanding of those changes.

4.2.1.1.3. Processes

Process modules are the core part of the system. There are two types of processing namely standard processing and sub-model processing. They are responsible for execution of entities that pass through them. The queue can actually be seen on the standard processing module since it has options for seizing and releasing of resources. There are many other entity related options like value added, non-value added, transfer, wait, etc which can customize the system as we want.

Sub-model processing modules are under the processing modules which enable larger and more complex logics in the system. In fact, with the presence of sub-model process, the system logic is processes mainly in the sub-model processes.

4.2.1.1.4. Variables

Variables or sometimes called as global variables are elements that provide information about the unknown values or characteristics of the system. Some variables are system generated such as queue at the *Process* module or number of busy servers while some are user defined like activity time, total time, etc. Variables are not related directly to any specific entity but rather they are influenced by the entire system.

We'll now showcase some of the variables we have used in our test model. This is subject to minor changes in the later part of the modified model made for tracing the critical path which is our primary goal. Also, an overview of Entities, Attributes and Variables will be given in the end in **Appendix A**. A variable list from Arena is shown below:

Table 1: Table from ARENA Showing Different Variables Used

Variable - Basic Process

	Name	Rows	Columns	Clear Option	Initial Values	Report Statistics
1	row_counter			System	0 rows	☐
2	num_of_activities			System	1 rows	☐
3	hold_activity	50		System	0 rows	☐
4	col_counter			System	0 rows	☐
5	found_a_1			System	0 rows	☐
6	dummy_counter			System	1 rows	☐
7	test	3		System	0 rows	☐
8	time used			System	0 rows	☐
9	start time			System	0 rows	☐
10	predecessors	50	15	System	0 rows	☐
11	row			System	0 rows	☐
12	p_index			System	0 rows	☐
13	col			System	0 rows	☐
14	Max_dependencies			System	1 rows	☐
15	dependency	50	50	System	0 rows	☐
16	current_activity	12		System	0 rows	☐
17	duration_matrix	50	3	System	0 rows	☐
18	show_duration	12		System	0 rows	☐
19	finish_time_resource	12		System	0 rows	☐

- **Statistical Variables**

Statistical variables are used to showcase the output statistics in the end of system run. These are used for statistical analysis like utilization rate and in our case we've used variables like *show_duration*, *finish_time_resource*, *start time*, etc which will help us compare the results with that of PERT's.

- **Condition/Logical Variables**

Condition variables are assigned to ensure that a condition is fulfilled and that the action should be taken on the entities. In our model, we have *Hold* module which is responsible for holding the entities until the system conditions are met and if true the entities are released. These variables are also known as logical variables using binary digits representing 0 to be true and 1 to be false. The actions of these logical variables are also supported by loops using looping logic.

- **Data Generation Variables**

These variables are used to generate data from input file over the system mainly attribute values of the entities. In the model, these attribute values will be checked and employed in correspondence with the logical variable to generate exact activity and dependency flow.

4.2.2. Activity Generation Sub Models

All the activities will be extracted from the Excel file in the beginning after the entities are created. As mentioned before, a dummy entity will represent the real activity and release it for further process. The accepted activities are then disbursed to *Hold* module through a *Separate* module where they are withheld until the conditions are satisfied for their run through the system.

A step- by- step explanation of the important intermediates of the model mainly the sub models which utilize the logic from reading data to processing the data in the exact way as desired will be given further from here. Below is the screenshot of the experimentation model:

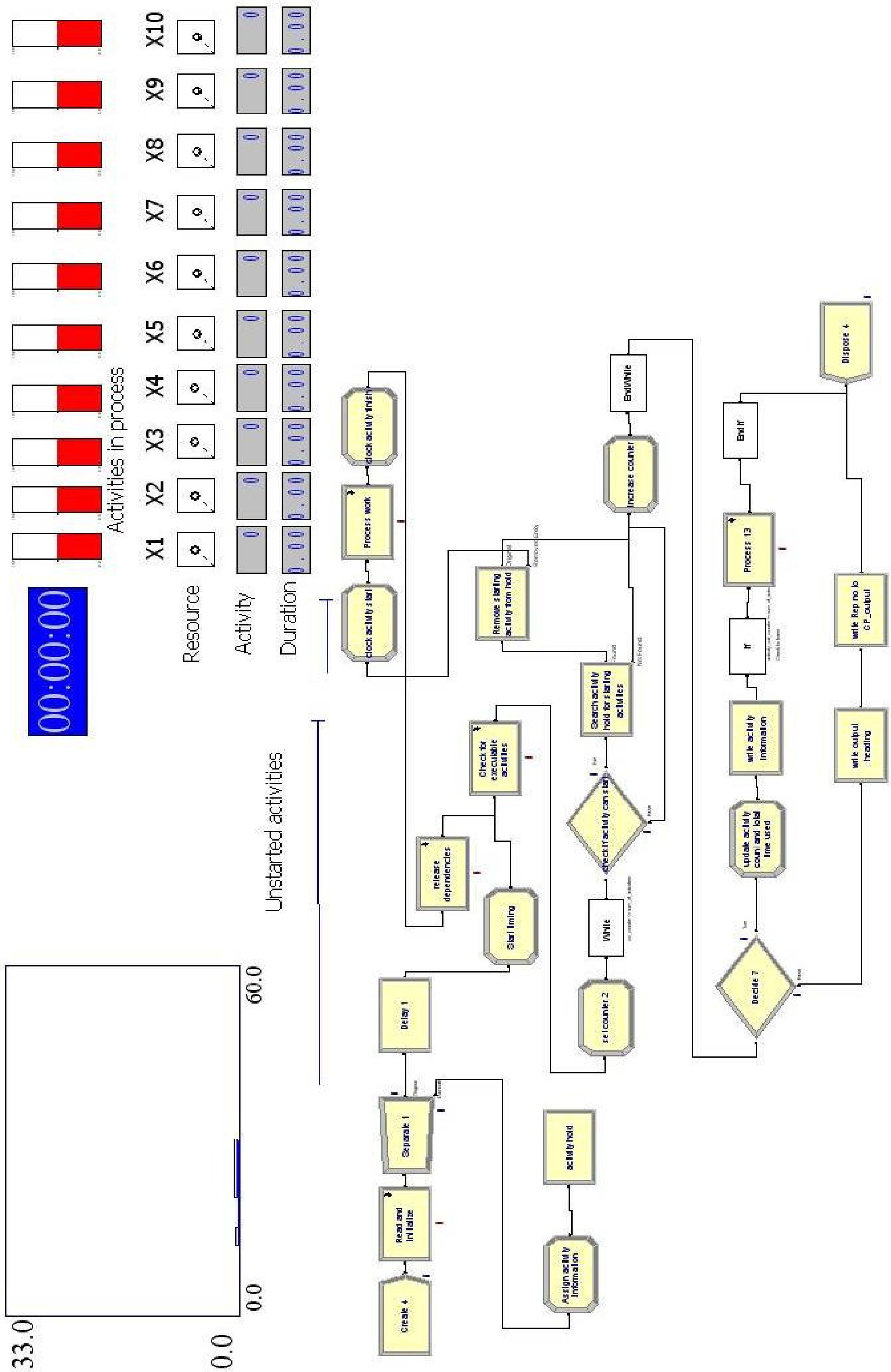


Figure 9: Model for Experimentation and Output Analysis

4.2.3. Data Extraction Sub Model

This sub model is necessary and meant for reading data from file into Arena. Under this sub model data is read using *ReadWrite* module which facilitates with file access to Arena.

The Read number of activities module reads the number for number of activities and then proceeds to reading the data for predecessors and duration for activities as long as the condition in the *While* loop that (row<= number of activities) is satisfied. It does a two dimensional screening of the data which is arranged in Excel in a way that it is compatible to what Arena works on. Looping command is used to ensure that all the data regarding durations and predecessors is covered in correspondence with number of activities.

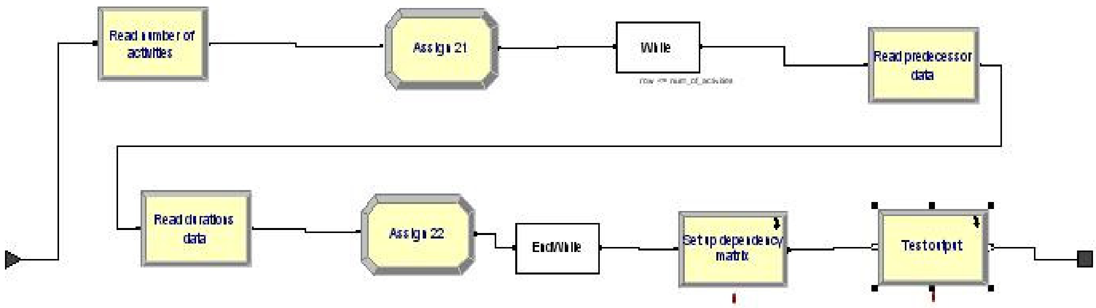


Figure 10: Sub Model for Reading Data for Activity Durations and Predecessors

4.2.4. Sub Model for Generating Dependency Matrix

Since the preliminary data about dependencies will be read from Excel and not directly from the matrix under the *Variable* section in Arena, we have to create and logical matrix in Arena based on the input data for predecessors from Excel file. It is too complicated or unknown in Arena about assigning dependency matrix and activity durations at the same time. So, basically, there will be another sub model called *Setup dependency matrix* to the sub model above. In this sub model, if the condition under the *Decide* model is true; then the dependencies will be assigned to a logical variable called *p_index*. The logical numbers 0 and

1 will be assigned into logical matrix *p_index* in correspondence with the predecessor data extracted from sub model in *Figure 10*.

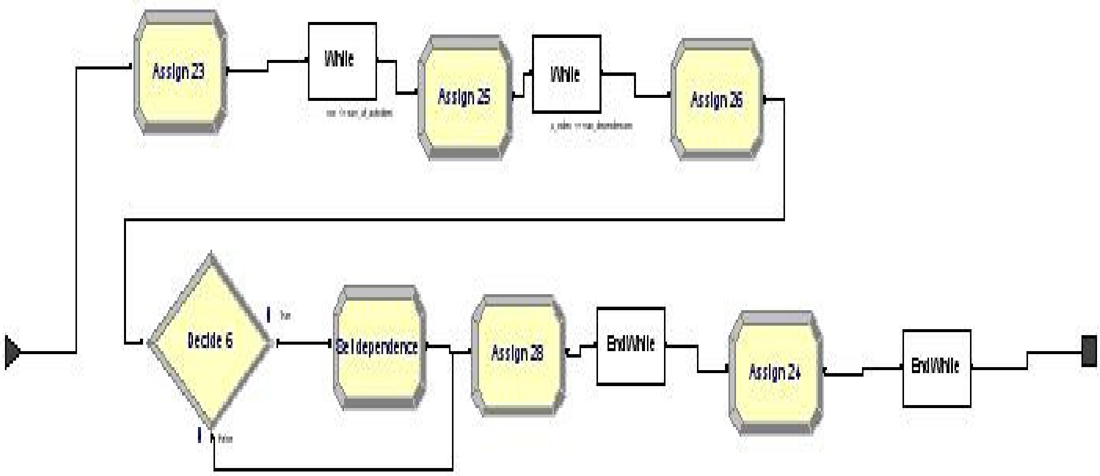


Figure 11: Sub Model for Creating Logical Dependency Matrix

4.2.5. Sub Model for Registering Output

Registration of dependency matrix and corresponding optimistic, normal and pessimistic durations is a mere validation method that has been employed by using this sub model. We should ensure that the logical matrix created in the sub model (*Figure 11*) is in accordance with the predecessor table and Gantt chart made using Microsoft Project. We can see the user defined information in an output file, validate it and proceed further on the system.

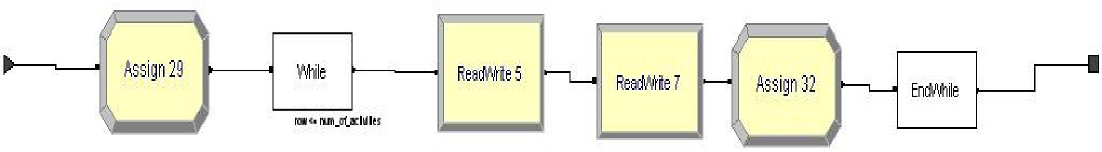


Figure 12: Sub Model for Validation of Data by Writing Output to a File

4.2.6. Sub Model for Non-Dependent Activities

For smooth processing of this sub model, the logic behind sub model in *Figure 11* is a pre-requisite. In *Figure 11* sub model, we create a dummy logical matrix in Arena which is the basis to starting up of non-dependent activities at first hand. But, there can also be a possibility that some of the dependencies to their predecessor and non-dependent activities need to be processed at the same time and hence, we have utilized 5 resources from X1 to X5, respectively. The matrix in discussion here is a two dimensional matrix, so, the logic here is to search both row and column for dependencies. The dependencies can be traced with binary variables 0 or 1. When the search is retrieving 1 that means that activity has a predecessor and it cannot start and is therefore held at the *Hold* module for time being. The two sub models below represent checking of executable activities and the activities that are identified as those that cannot be executed are registered and the looping continues to look for executable and non-executable activities.

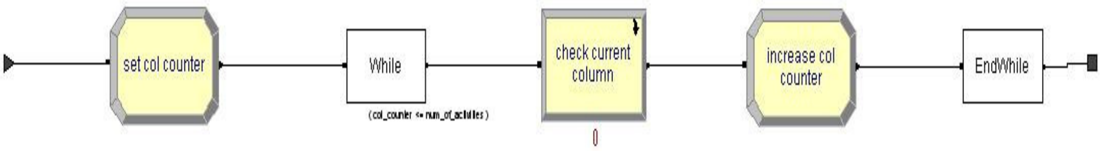


Figure 13(A): Sub Model for Executing Activities without Dependencies

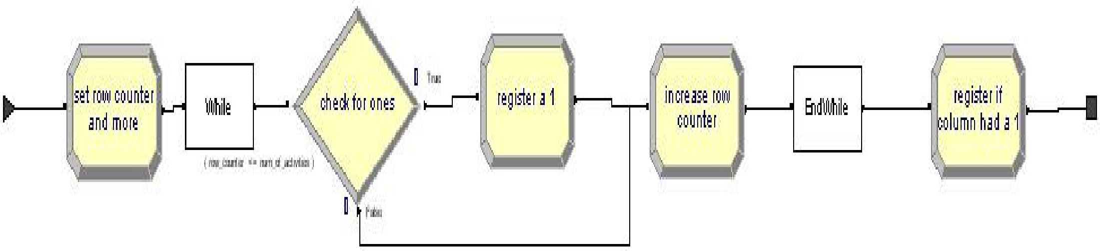


Figure 13(B): Sub Model for Executing Activities without Dependencies

4.2.7. Sub Model for Releasing Dependencies

Henceforth, a *Decide* module is placed mainly for checking if the dependencies can start. It's basically the second stage of system run and primarily meant for dependencies. After the executable activities have been identified, they are removed from the *Hold* module and are processed. After the execution of those activities, their dependencies are released and the same procedure is followed on them as described from *Figure 13 (A) & 13 (B)* until the *Decide* module where checking is done whether the dependencies released on the basis of one of their processed predecessors is able to start or not. For instance, there can be case when activity 4 is dependent on both 1 & 7. When activity 1 is done, dependent activity 4 is released but after further checking, the matrix at 4-7 retrieves a non-zero value which is 1 which means that activity 4 is also dependent on 7 and therefore is withheld at *Hold* queue.

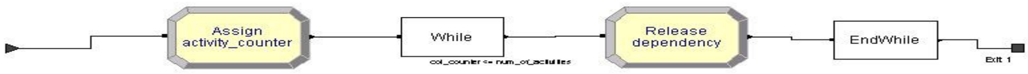


Figure 14: Sub Model Responsible for Releasing Dependencies

4.3. Animation

A bit of animation is also provided to have first hand visual confirmation of what is going on in the system. The animation is very elegant and provides with what we intend to look for. We have animation for resources, duration and activities.

For resources, an option from tool bar is chosen which guides you to resource animation option and from there we have to check seize option that enables us to see the status of the resource used whether busy or idle. Green colour signals for busy status.

Animation for activities and durations is quite sophisticated and easily understandable. They stand for the duration taken by every activity under a particular resource (the signal boxes are placed one below another). There is also a tank like animation which empties as the time on certain activity under certain resource is being utilized.

4.4. Data Provision and Description

Our objective of this test is to defend the use of simulation and to see that the output regarding estimated time and critical path actually deviate.

PERT uses the probabilistic approach to calculate three different times namely the optimistic time, pessimistic time and the estimated time. We'll run the model with 1000 replications which will strengthen our claims about critical paths acquired from Arena. Three different cases namely 'Case A' with optimistic and pessimistic times being 60% and 200% of normal, 'Case B' with optimistic and pessimistic times being 80% and 150% of normal and 'Case C' with optimistic and pessimistic times being 95% and 110% of normal on the model.

Through PERT we can see an obvious critical path using Gantt chart. Uncertainty is also covered in PERT and then only estimates are made and the range of uncertainty lies from Optimistic-Normal-Pessimistic durations.

But as you see the later part, we should be able to reveal that simulation can give distinct results and alternatives after we replicate the model with uncertainty in terms of probabilistic distribution. We'll find a reason to comment about level of uncertainty, re-planning approaches in terms of scheduling and re-allocation of resources.

As shown in both the tables, first and second column represent activity number and name; third, fourth and fifth represent optimistic, normal and pessimistic times; sixth column represents expected time calculated using the formula for expected time in PERT and the last column stands for predecessors.

The dependency table will remain common for all the three cases. The Gantt charts for the three cases are presented below:

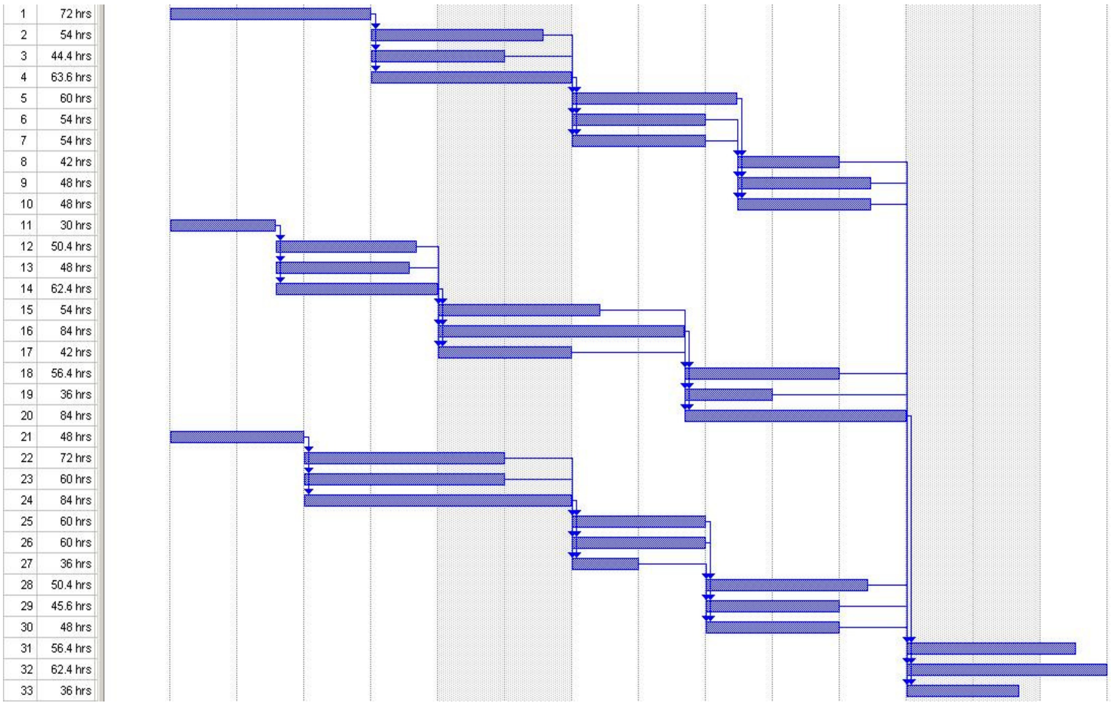


Figure 15: Gantt chart for 'Case A' from PERT

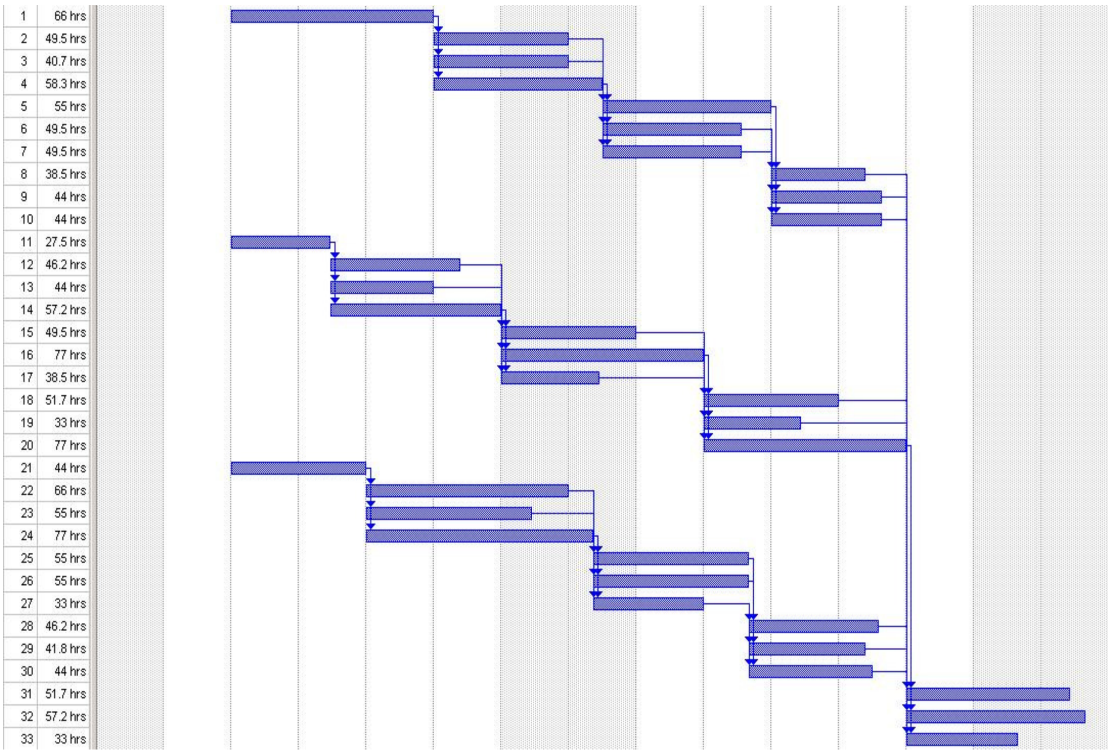


Figure 16: Gantt chart for 'Case B' from PERT

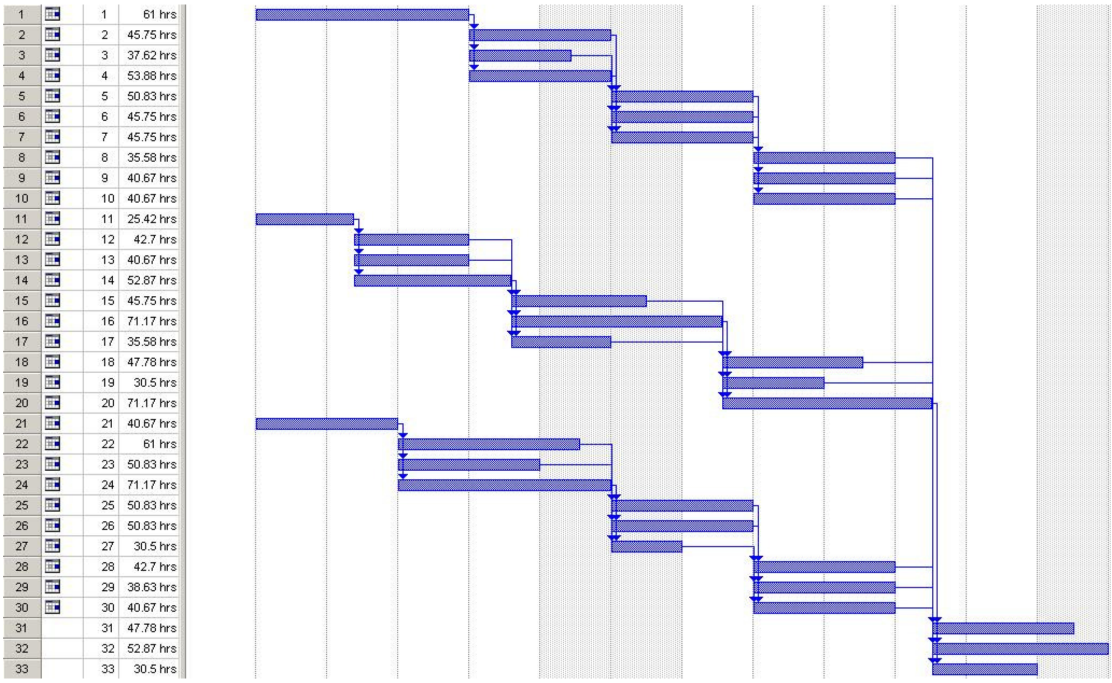


Figure 17: Gantt chart for 'Case C' from PERT

The Gantt charts imply almost the same image in all of the three cases except for some minor variation caused due to percent changes in optimistic and pessimistic times. The Gantt charts will also showcase an obvious critical path for all of the three cases. The dependency table to be read by simulation model is common for all the three cases and is presented in **Appendix B**.

Table 2: Data for 'Case A' with Optimistic and pessimistic Durations as 60% and 200% of Normal Durations

Act#	Optimistic	Normal	Pessimistic	Te / Expected	Predecessors
1	36	60	120	72	
2	27	45	90	54	1
3	22,2	37	74	44,4	1
4	31,8	53	106	63,6	1
5	30	50	100	60	4;2;3
6	27	45	90	54	4;2;3
7	27	45	90	54	4;2;3
8	21	35	70	42	7;5;6
9	24	40	80	48	7;5;6
10	24	40	80	48	7;5;6
11	15	25	50	30	
12	25,2	42	84	50,4	11
13	24	40	80	48	11
14	31,2	52	104	62,4	11

15	27	45	90	54	14;12;13
16	42	70	140	84	14;12;13
17	21	35	70	42	14;12;13
18	28,2	47	94	56,4	17;15;16
19	18	30	60	36	17;15;16
20	42	70	140	84	17;15;16
21	24	40	80	48	
22	36	60	120	72	21
23	30	50	100	60	21
24	42	70	140	84	21
25	30	50	100	60	24;22;23
26	30	50	100	60	24;22;23
27	18	30	60	36	24;22;23
28	25,2	42	84	50,4	27;25;26
29	22,8	38	76	45,6	27;25;26
30	24	40	80	48	27;25;26
31	28,2	47	94	56,4	8;9;10;18;19;20;28;29;30
32	31,2	52	104	62,4	8;9;10;18;19;20;28;29;30
33	18	30	60	36	8;9;10;18;19;20;28;29;30

Table 3: Data for 'Case B' with Optimistic and pessimistic Durations as 80% and 150% of Normal Durations

Act#	Optimistic	Normal	Pessimistic	Te / Expected	Predecessors
1	48	60	90	66	
2	36	45	67,5	49,5	1
3	29,6	37	55,5	40,7	1
4	42,4	53	79,5	58,3	1
5	40	50	75	55	4;2;3
6	36	45	67,5	49,5	4;2;3
7	36	45	67,5	49,5	4;2;3
8	28	35	52,5	38,5	7;5;6
9	32	40	60	44	7;5;6
10	32	40	60	44	7;5;6
11	20	25	37,5	27,5	
12	33,6	42	63	46,2	11
13	32	40	60	44	11
14	41,6	52	78	57,2	11
15	36	45	67,5	49,5	14;12;13
16	56	70	105	77	14;12;13
17	28	35	52,5	38,5	14;12;13
18	37,6	47	70,5	51,7	17;15;16
19	24	30	45	33	17;15;16
20	56	70	105	77	17;15;16
21	32	40	60	44	
22	48	60	90	66	21
23	40	50	75	55	21
24	56	70	105	77	21
25	40	50	75	55	24;22;23

26	40	50	75	55	24;22;23
27	24	30	45	33	24;22;23
28	33,6	42	63	46,2	27;25;26
29	30,4	38	57	41,8	27;25;26
30	32	40	60	44	27;25;26
31	37,6	47	70,5	51,7	8;9;10;18;19;20;28;29;30
32	41,6	52	78	57,2	8;9;10;18;19;20;28;29;30
33	24	30	45	33	8;9;10;18;19;20;28;29;30

Table 4: Data for 'Case C' with Optimistic and pessimistic Durations as 95% and 110% of Normal Durations

Act#	Optimistic	Normal	Pessimistic	Te / Expected	Predecessors
1	57	60	66	61,00	
2	42,75	45	49,5	45,75	1
3	35,15	37	40,7	37,62	1
4	50,35	53	58,3	53,88	1
5	47,5	50	55	50,83	4;2;3
6	42,75	45	49,5	45,75	4;2;3
7	42,75	45	49,5	45,75	4;2;3
8	33,25	35	38,5	35,58	7;5;6
9	38	40	44	40,67	7;5;6
10	38	40	44	40,67	7;5;6
11	23,75	25	27,5	25,42	
12	39,9	42	46,2	42,70	11
13	38	40	44	40,67	11
14	49,4	52	57,2	52,87	11
15	42,75	45	49,5	45,75	14;12;13
16	66,5	70	77	71,17	14;12;13
17	33,25	35	38,5	35,58	14;12;13
18	44,65	47	51,7	47,78	17;15;16
19	28,5	30	33	30,50	17;15;16
20	66,5	70	77	71,17	17;15;16
21	38	40	44	40,67	
22	57	60	66	61,00	21
23	47,5	50	55	50,83	21
24	66,5	70	77	71,17	21
25	47,5	50	55	50,83	24;22;23
26	47,5	50	55	50,83	24;22;23
27	28,5	30	33	30,50	24;22;23
28	39,9	42	46,2	42,70	27;25;26
29	36,1	38	41,8	38,63	27;25;26
30	38	40	44	40,67	27;25;26
31	44,65	47	51,7	47,78	8;9;10;18;19;20;28;29;30
32	49,4	52	57,2	52,87	8;9;10;18;19;20;28;29;30
33	28,5	30	33	30,50	8;9;10;18;19;20;28;29;30

4.5. Test Model Verification

We need to verify that our model responds in the same way as we intent it to. For instance, we need to ensure that there are no errors while running the model. There have been instances when Arena revert an error message that something has not been defined and so on. So, we have to run the model for specific time duration to ensure error free run.

In our case verification will be done on the data that we are running the model with. In one of the sub models above that handles the data, we have made adjustment for verifying the data out in an output file where we can tally the data used in the input with that of the data acquired from output file. These two need to match in order to ensure that Arena actually uses the data we have given it and that it performs the same way as we have desired.

We have made separate dependency table based on predecessors in **Appendix B** which will be used as input files for generating dependencies along with three different types of durations mainly optimistic, normal and pessimistic which are taken from *Tables 2, 3 and 4.*

The system is being run with 1000 replications and replication length to be infinite. After running the system, we could verify that the data used by the system is the same as we fed in. It means that the model uses the data correctly without causing any errors. We also found that the model ran smoothly for specified number of replications without showing any symbolic errors like undefined attributes, etc. We could also confirm from our observations and animation that the system generates accurate dependencies as fed in the data but obviously there will be deviation in the durations since durations in our model are dealt by Arena with **Triangular Distribution**.

4.6. Test Model Validation and Output Experimentation

Validation is a method of ensuring whether the model functions in the same way as the real system. We actually don't have a real benchmark able system with which we can compare our system results but our results can be seen from experiment point of view and can possibly be interpreted into a real system going on at Ulstein.

The presumed data is built with an attempt to make it resemble a complex reality which displays complex situations with substantial uncertainty on hand. Some additional sub models processing modules have been added to get output regarding critical path to an output file.

The present experimental data includes several activities moving at the same time and they even consume almost the same time length before completion. The data has been designed in such a way that it shows how the critical path deviates certain percent of the times from the actual one acquired from PERT.

Some special features in the model include provisions to see finish order for activities with finishing durations for the same in the broken down order for number of replications the model is run for. We can also see what critical path we get for each replication. The critical paths can be seen to be varying. From PERT, we can see an obvious critical path which is 11-14-16-19-32 in all of the three cases.

The total execution time from PERT in all the three cases with critical path as 11-14-16-19-32 is as follows:

Table 5: Result of Three Cases Acquired from PERT

	Case A	**Case B**	**Case C**
Total Expected Execution Time	**322.8**	**295.9**	**273.5**

The uncertainty level varies from case to case. From the PERT results, we can observe that when optimistic time and pessimistic time are 60% and 200% of the normal, the total execution time is comparatively longer than other two cases. The reason for this, with the use of **Triangular Distribution**, 200% of inclination towards the pessimistic times in all the critical activities makes the expected time move rightwards from mean which in our case is the normal time. So, when the critical path is constant as seen in PERT, control over uncertainty will mind the critical path in terms of total project duration. Practically, it might be very difficult to have expected times to be in the range of optimistic and normal times but since there is always this key element of uncertainty in every one-of-a-kind project development, there are prospects for narrowing down the uncertainty curve based on forecasts or pre-emptive decision making tools. The narrowing down of uncertainty can then be directly interpreted in terms of currency/cost reduction which is the basis for profit maximization.

For the reasons above, we perceived simulation as a profile for showing feasibility that there could be alternatives to critical path and duration figures obtained from PERT.

Before moving to the analysis of varying critical paths, we would like to display the variation in execution times for the three cases obtained from simulation model in Arena. The execution times for 'Case A' and 'Case B' are in contrast to those in PERT except 'Case C' which released almost the same result as in PERT. The table is shown below and the figures can be compared with figures in *Table 5*.

Table 6: Result of Three Cases from Simulation Model

	Case A	**Case B**	**Case C**
Total Average Execution time	**358.63**	**306.78**	**273.65**
Half Width	**1.60**	**0.87**	**0.24**
Minimum Execution Time	**282.33**	**268.56**	**263.04**
Maximum Execution Time	**441.53**	**355.27**	**285.91**

'Case C' has the result close to PERT because 'Case C' has optimistic and pessimistic times to be 95% and 110% of normal; the level of uncertainty is very low compared to other two cases. We can assume that the system in 'Case C' is in a way deterministic and works as per plans.

Half width represents optimistic and pessimistic variation over the average result (Half width $\pm$ Total Average Execution Time). The average execution time is taken from all the 1000 replications we have run the model for. Minimum and maximum execution times represent the best and the worst case scenarios for the model. In other words, these are the upper and lower limits for durations.

Table 7: Comparison Between PERT and Arena Results

	Case A	Case B	Case C
Total Expected Execution Time by PERT	322.8	295.9	273.5
Total Average Execution Time by Arena	358.63	306.78	273.65
% Change	(-) 11. 09	(-) 3.67	(-) 0.055

From the above comparison, we can see how different Arena behaves from PERT except in 'Case C'. This is an evidence for us to explore more into how differently the critical path behaves from the one in PERT. We have run the model with 1000 replication on each of the cases, so, there is great possibility of variations in critical paths as the replications proceed. In order to know how many different critical paths are generated by the model, we have had to make provisions in the sub model for extracting the output to a file.

With this provision, we can figure out how many times each of the activities have occurred on the critical path along with its occurrence rate. This leads us to know how many critical activities are there on the project and there relationship with uncertain situation since all the three different cases will give different occurrence rates on the critical path for the critical activities.

The assessment of the above will also lead us to assess the occurrence rate for key critical paths that have occurred several times during the 1000 replications. With this information, we can draw a critical path network by assigning priority (in %) from the most important to the least important among the sorted critical paths.

5. Discussion on Experiment Output

The total execution times from the three cases look quite satisfactory and feasible in relation to PERT. The execution times acquired from Arena are on the pessimistic side to those obtained from PERT. As a strong focus of our thesis is not to achieve better output than PERT but to focus on the critical activities and how they influence the critical path with the use of simulation as well their influence on the planning methodologies for the project.

The good thing is that we managed to see variation in the critical path as expected. As per PERT, the critical path is constant in all the cases but Arena gives us various critical paths. Obviously, the critical path from PERT will always be there in Arena results also on more occasions than others but not 100% as in PERT except in 'Case C' (due to less uncertainty), in fact the main critical path 11-14-16-19-32 is achieved not more than 30% of the times in 'Case A' and 'Case B'. So, it means there can be other alternatives to the main critical path from planning point of view.

Three activities 1, 11 and 21 are the most important since they are the first to start among the activities from their dependency branches and also that these three activities are not inter-dependent. In 1000 replications in 'Case A', activities 1, 11 and 21 which are the starting activities for their respective work trees occur almost equally on the critical paths. There are a total of 108 different critical paths. The ratio of their occurrence to each other is almost 1:1:1 which means at least 30% of the times these activities occur on the critical path.

In 'Case B', there are 68 different critical paths with main activities 1, 11 and 21 occurring 24.2%, 52.6% and 23.1% of the times on the critical paths, respectively. In 'Case C', there are just 4 different critical paths with main activities 1, 11 and 21 occurring 0.2%, 99.6% and 0.1% of the times on the critical paths, respectively. 'Case C' gives almost the same critical path as PERT 99.6% of the times due to less uncertain nature of the activity durations which enables the model to identify the same critical path always. Tables of all the three cases consisting of critical path number, critical path, number of occurrences and occurrence rate are given in **Appendix D**. Charts presenting the occurrence rate of all the activities on the critical path are also given in **Appendix C** for reference.

Except 'Case C', the other two cases have many different critical paths which are difficult to accommodate in the tree diagram that we intend to make. So, we'll consider only those critical paths with occurrence rate at least 1% which will make it easy and less cumbersome to draw. With at least 1% occurrence criteria, we have identified 30 critical paths in 'Case A', 25 in 'Case B' and just one in 'Case C' which is very obvious. 'Case C' has a total of 4 different critical paths which will be drawn in full.

5.1. 'Case A' Critical Path Output

'Case A' with optimistic and pessimistic times to be 60% and 200% of normal times proved to be a very good example of a project with high uncertainty. In this case, the critical path deviates 88% of the times from the critical path similar to PERT's (11-14-16-20-32) and occurring just 12% of the times. The other major critical path following is 11-14-16-20-31 with occurrence of 7.8% and also several more which have occurrence rate that is 1/3rd of 11-14-16-20-32. The occurrences of critical paths are presented in *Figure 18* with the numbers over the bars representing the critical path numbers which can be verified from **Appendix D**.

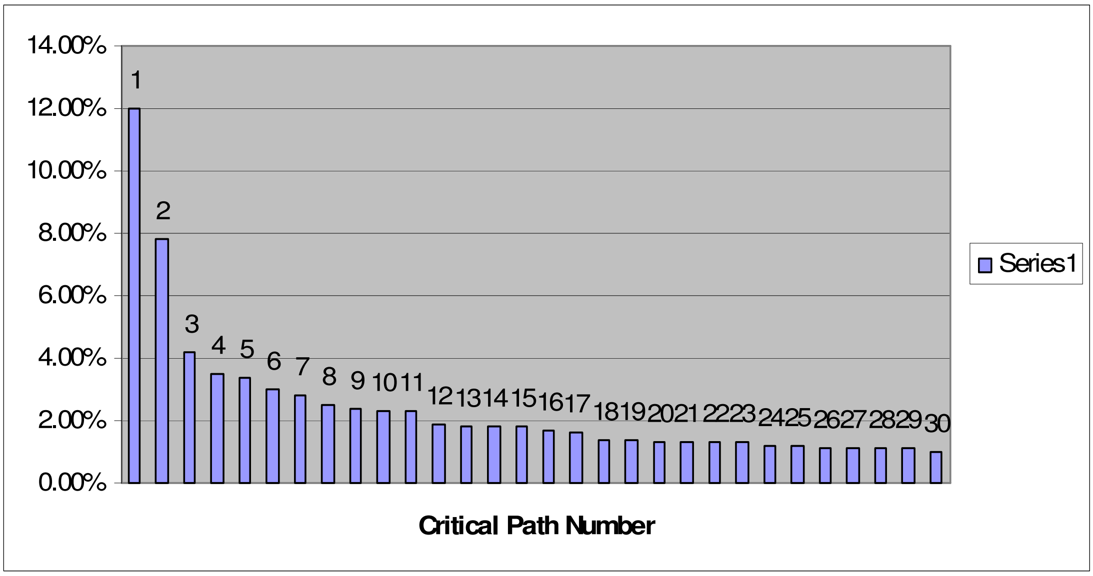
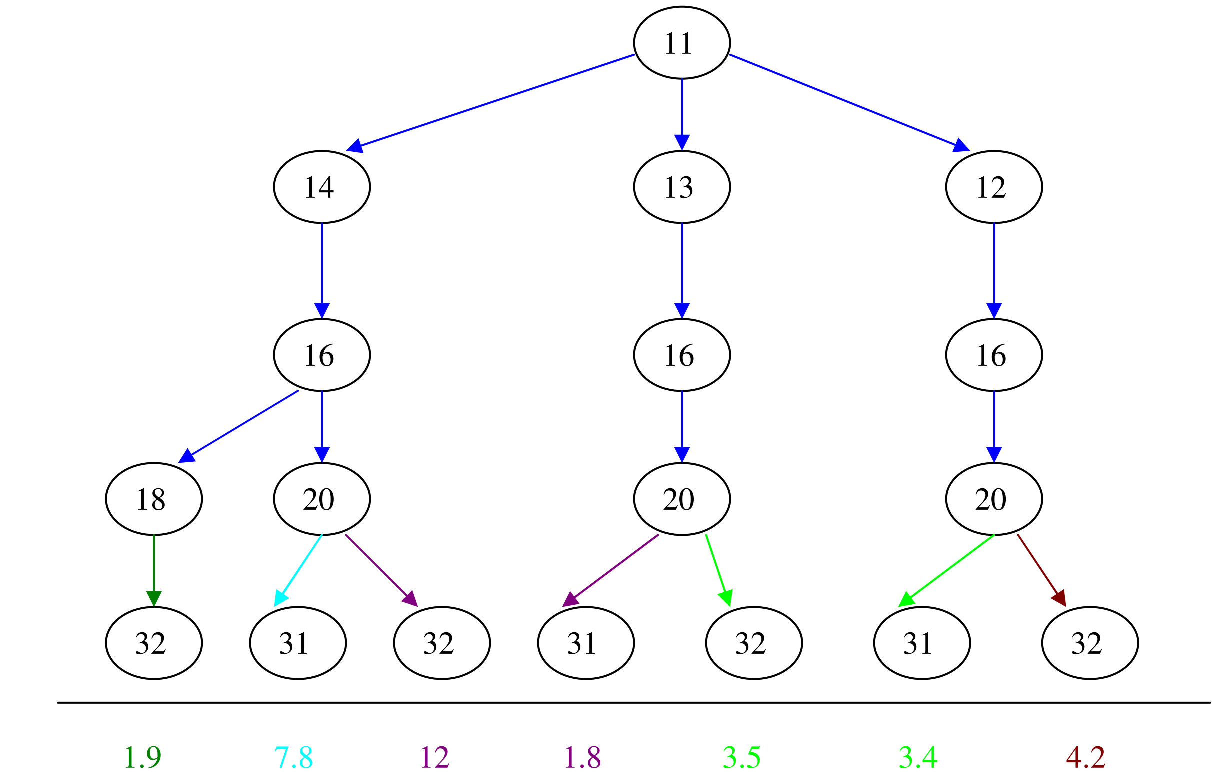
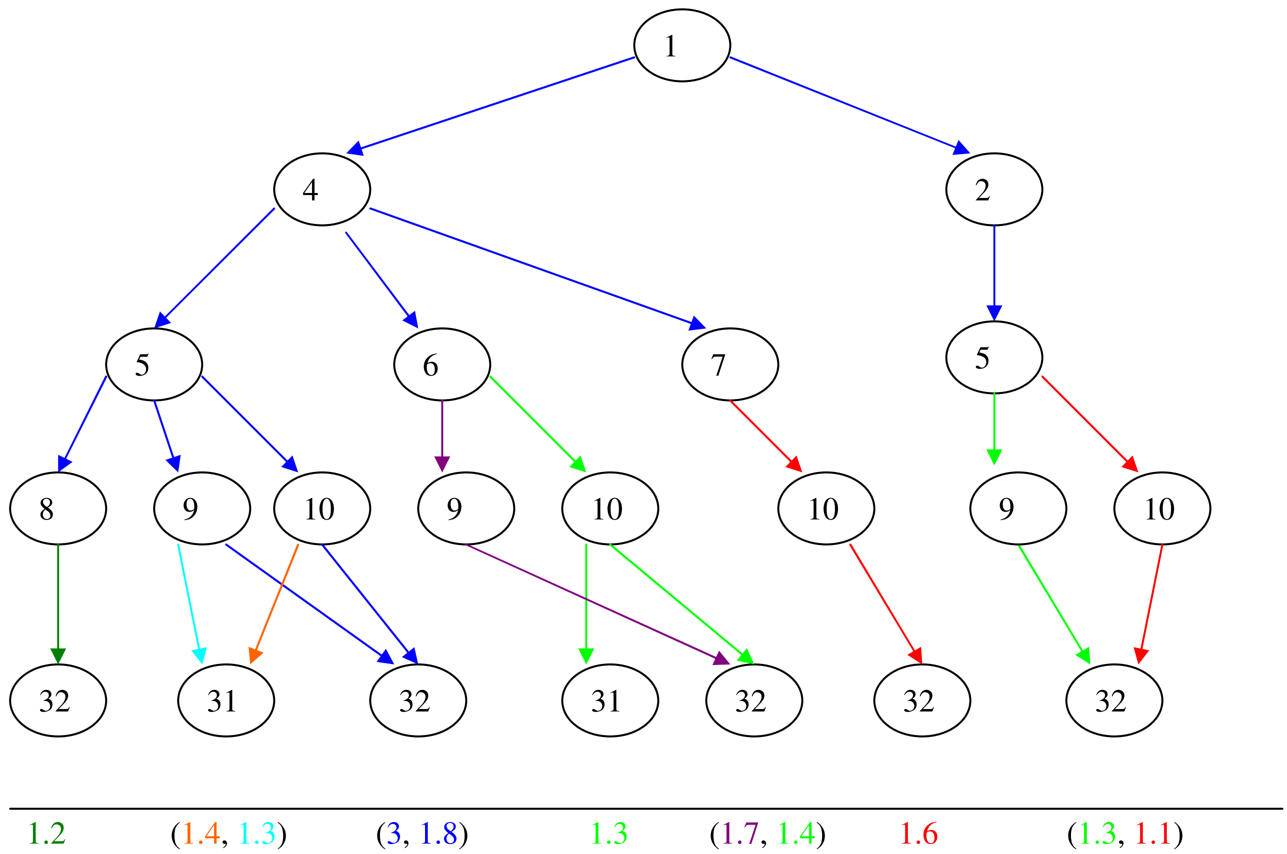


Figure 18: Bar Diagram for Occurrences of Critical Paths in 'Case A'

This various effect is the cause of huge uncertainty from normal as well as less restriction on the model which could have been enforced by using conditions/constraints especially resource based constraints. In projects, the uncertain nature has a direct impact from first hand planning and has an equal impact on re-planning methods after the uncertainty is identified. That's the reason why we need a pre-emptive tool. For instance, we can compare critical path number 1, 2, 3, 6 and 7. All these include critical paths with either of the three initial main activities. From planning perspective, inability to offset uncertainty on one of the main critical paths drives the focus of the team to complete other parts of the project that are not inter-dependent on the one with uncertainty until a certain point. This makes other parts of the project the critical path from time to time. This claim can be related to occurrence rates in critical path numbers 2, 3, 6 and 7 which are driven just due to uncertainty on critical path 11-14-16-20-32. Critical path number 2 (11-14-16-20-31) implies that critical path number 1 (11-14-16-20-32) failed to meet pre-requisites for activity number 32 7.8% of the times and vice versa for other variations.

A critical path layout of 'Case A' including all the 30 critical paths is drawn below. The numbers below the line represent occurrences of the critical paths in percentage (%).


11
14
13
12
16
16
16
18
20
20
20
32
31
32
31
32
31
32
1.9
7.8
12
1.8
3.5
3.4
4.2


1
4
2
5
6
7
5
8
9
10
9
10
10
9
10
32
31
32
31
32
32
32
1.2
(1.4, 1.3)
(3, 1.8)
1.3
(1.7, 1.4)
1.6
(1.3, 1.1)

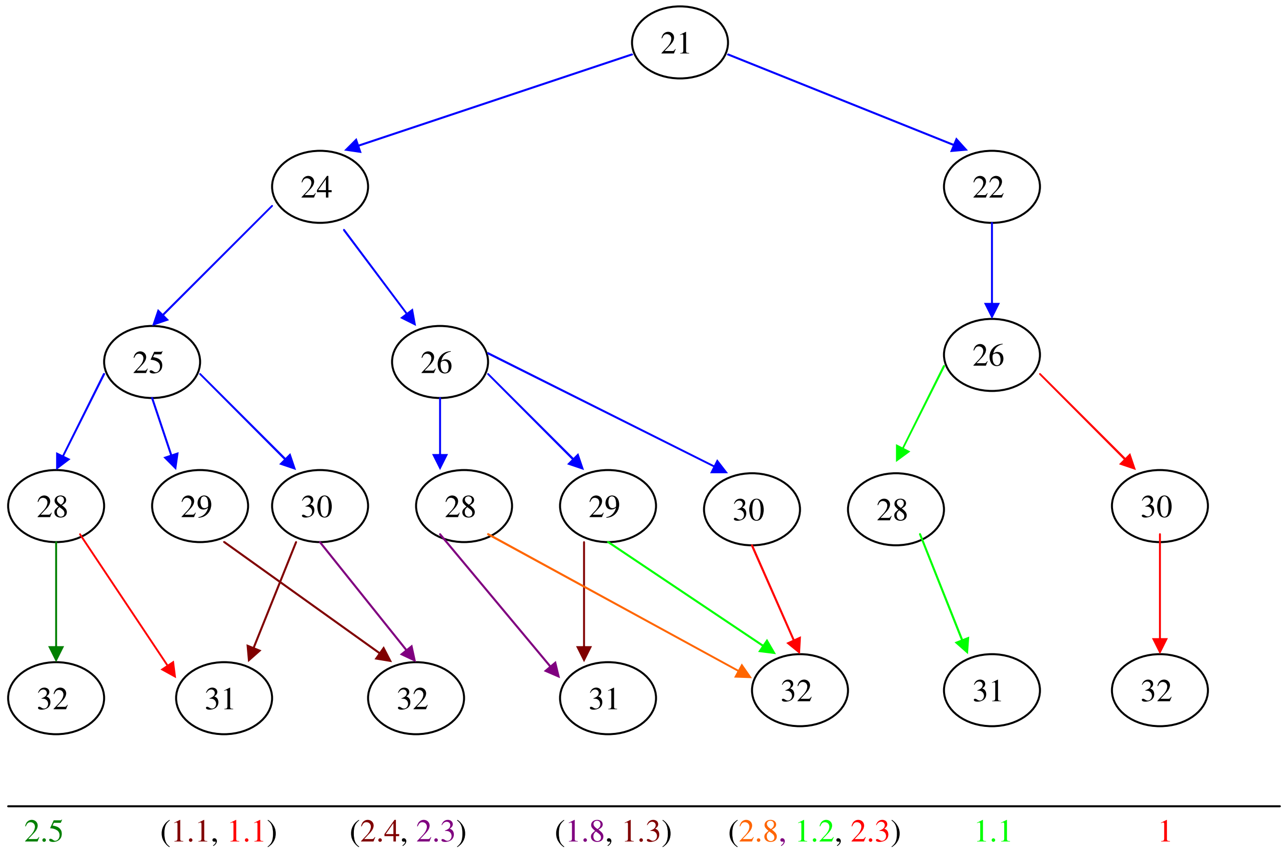


Figure 19: Critical Path Solution Layout for 'Case A'

5.2. 'Case B' Critical Path Output

'Case B' with optimistic and pessimistic times to be 80% and 150% of normal times also proved to be a very good example of a project with medium uncertainty. In this case, the critical path deviates 70% of the times from the critical path similar to Pert's (11-14-16-20-32) and occurring 30% of the times with an improvement of 18% over 'Case A'. The other major critical path following is 11-14-16-20-31 with occurrence of 12.8% and also 1-4-5-9-32 and 21-24-26-28-32 occurring 3.8% and 3.4% of the times, respectively. The occurrences of critical paths are presented in *Figure 20 below* with the numbers over the bars representing the critical path numbers which can be verified from **Appendix D**.

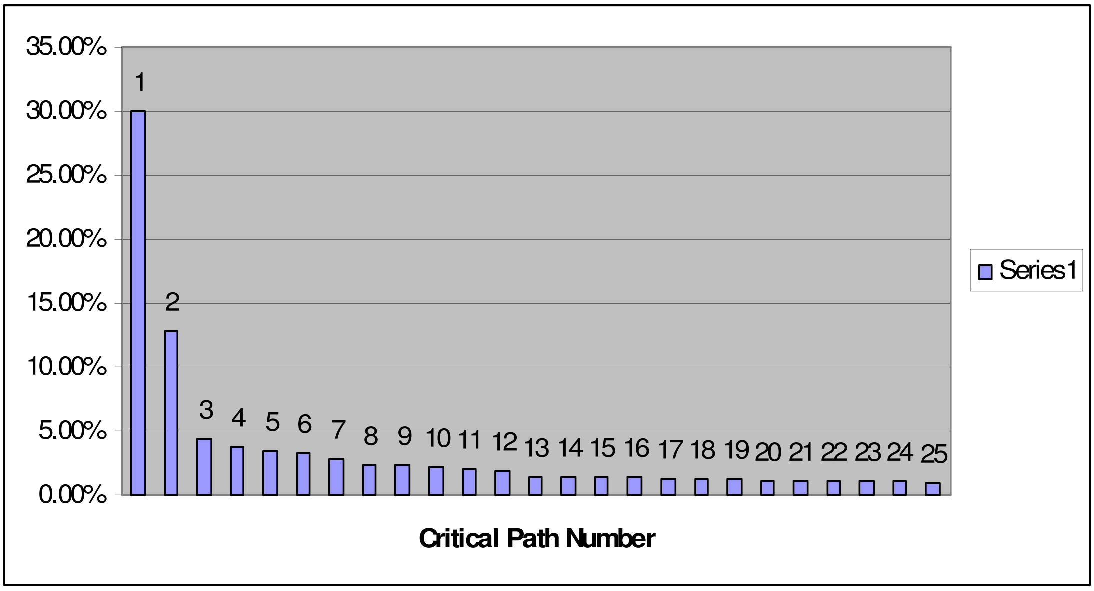
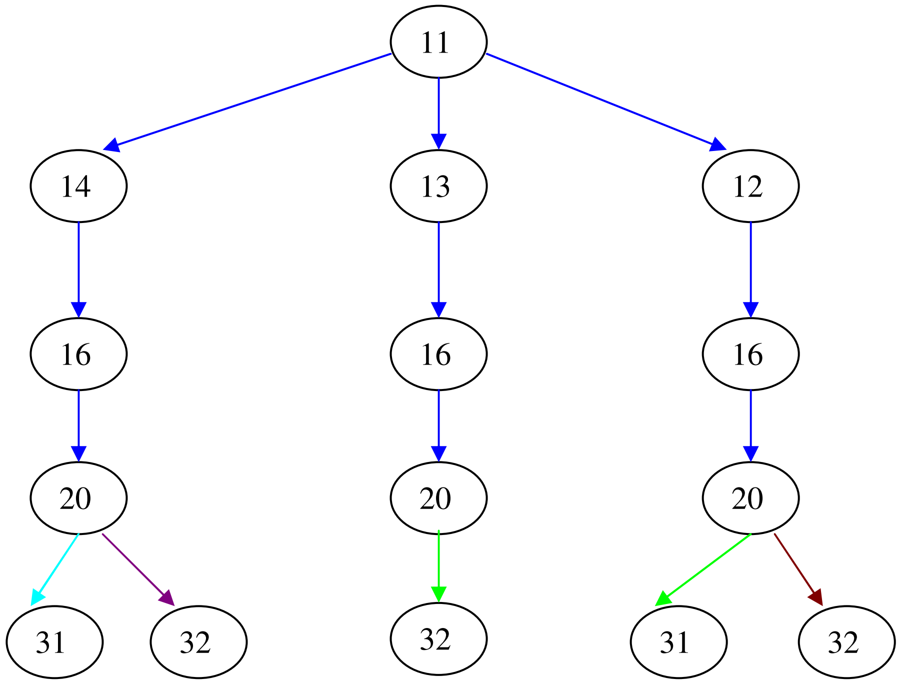


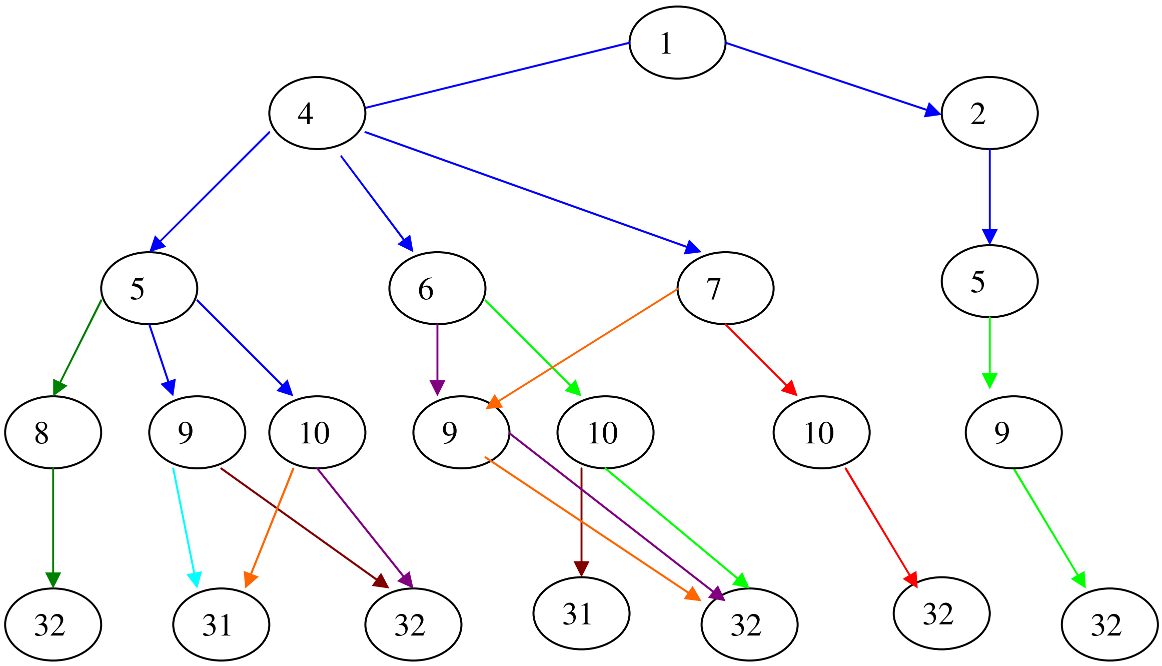
Figure 20: Bar Diagram for Occurrences of Critical Paths in 'Case B'

As the uncertainty is reduced, the weight age of the main critical path rises up by 18% from last case. Even the ratio of occurrence of critical path number 1 to number 2 has gone more than double from the last case. 'Case A' and 'Case B' can be good samples of extreme uncertainty in which maintenance of pre-requisites (kitting) can be very helpful in averting internal uncertainties.

A critical path layout of 'Case B' including all the 25 critical paths is drawn below. The numbers below the line represent occurrences of the critical paths in percentage (%).



12.8 30 2.2 2.1 4.4



1.1 (1.2, 1.4) (3.8, 3.3) 1.1 (1.4, 1.2, 1.1) 1.3 1

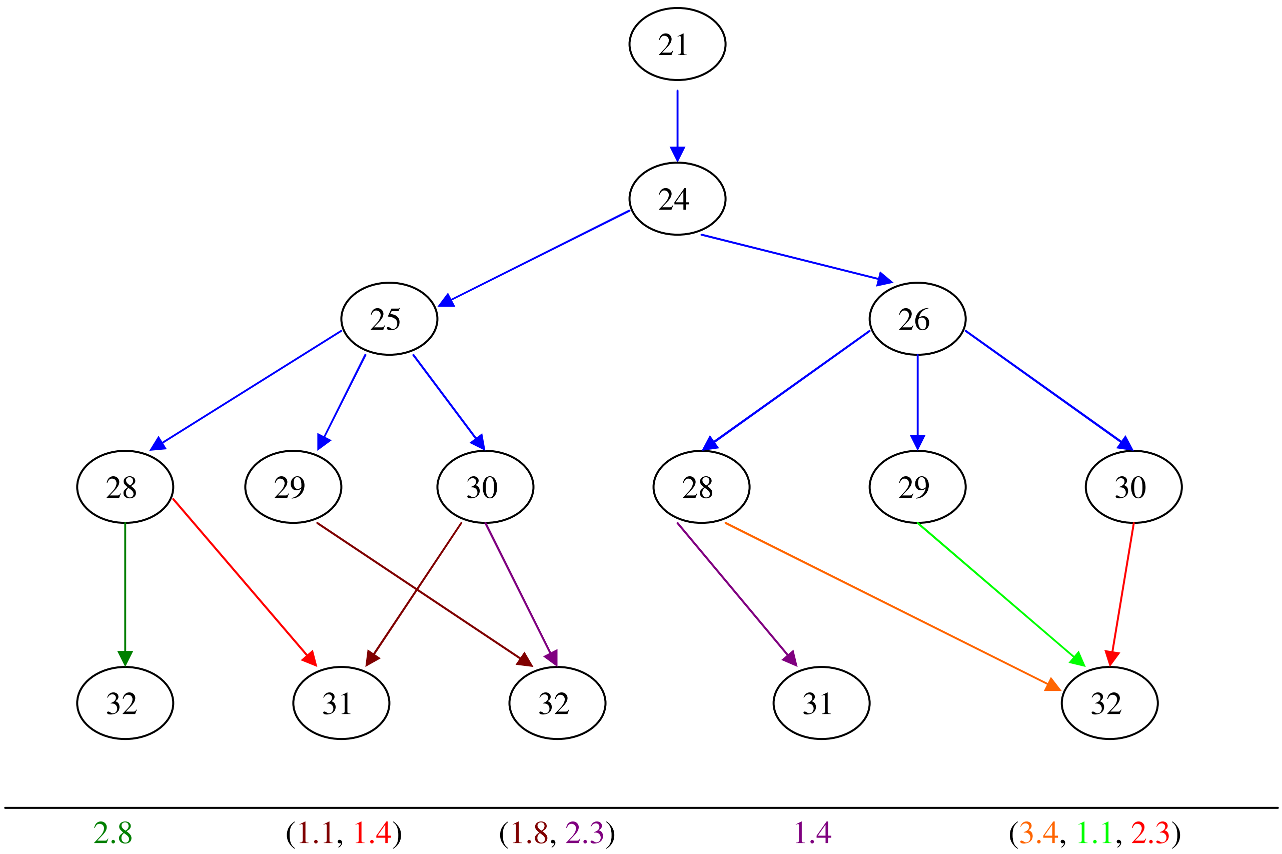


Figure 21: Critical Path Solution Layout for 'Case B'

5.3. 'Case C' Critical Path Output

'Case C' with optimistic and pessimistic times to be 95% and 110% of normal times completes our exemplified trio with argument regarding uncertainty and its influence on the project time. In this case, the critical path deviates less than 1% of the times with the critical path similar to Pert's (11-14-16-20-32) and occurring 99.2% of the times. The occurrences of critical paths are presented in *Figure 22* with the numbers over the bars representing the critical path numbers which can be verified from **Appendix D**.

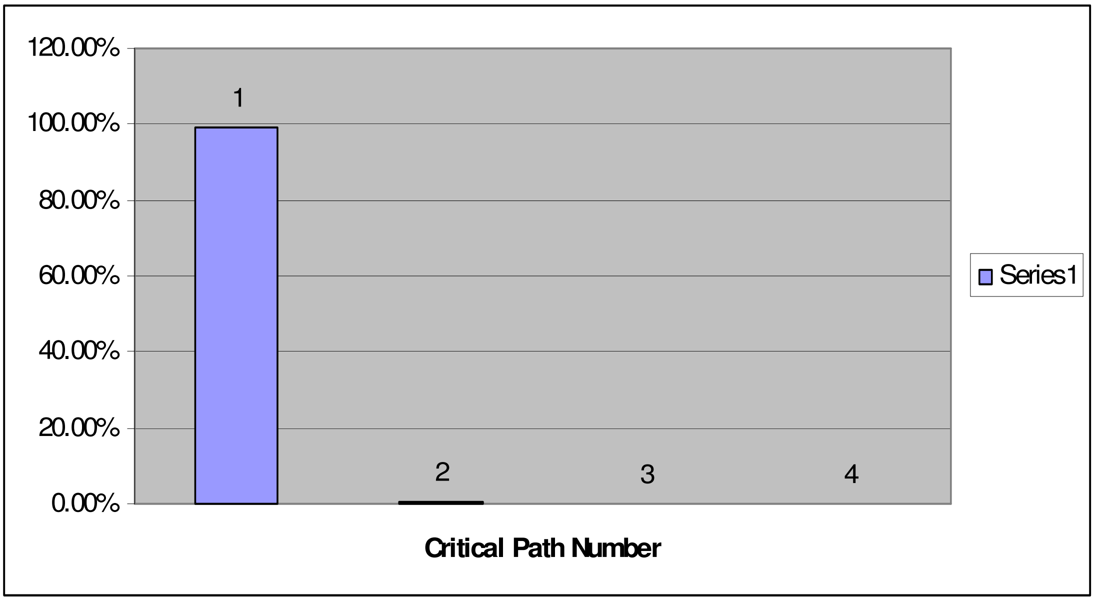


Figure 22: Bar Diagram for Occurrences of Critical Paths in 'Case C'

This case is very straight forward with low uncertainty compared to other cases. The total execution time achieved is also the same as from PERT. It means that this case represents an ideal situation as in PERT. We can also perceive from the result of this case and the former ones that critical path is actually a bottleneck in circumstances of uncertainty. Execution failure of any of the activities on the critical path on time spoils the entire critical path and can thus spoil the project length since the activities that are dependent on the critical ones are thereby delayed due to delay on the critical path. But, there are also possibilities of re-planning the uncertain situation by effective use of planning tools.

A critical path layout of 'Case C' including all the 4 critical paths is drawn below. The numbers below the line represent occurrences of the critical paths in percentage (%).

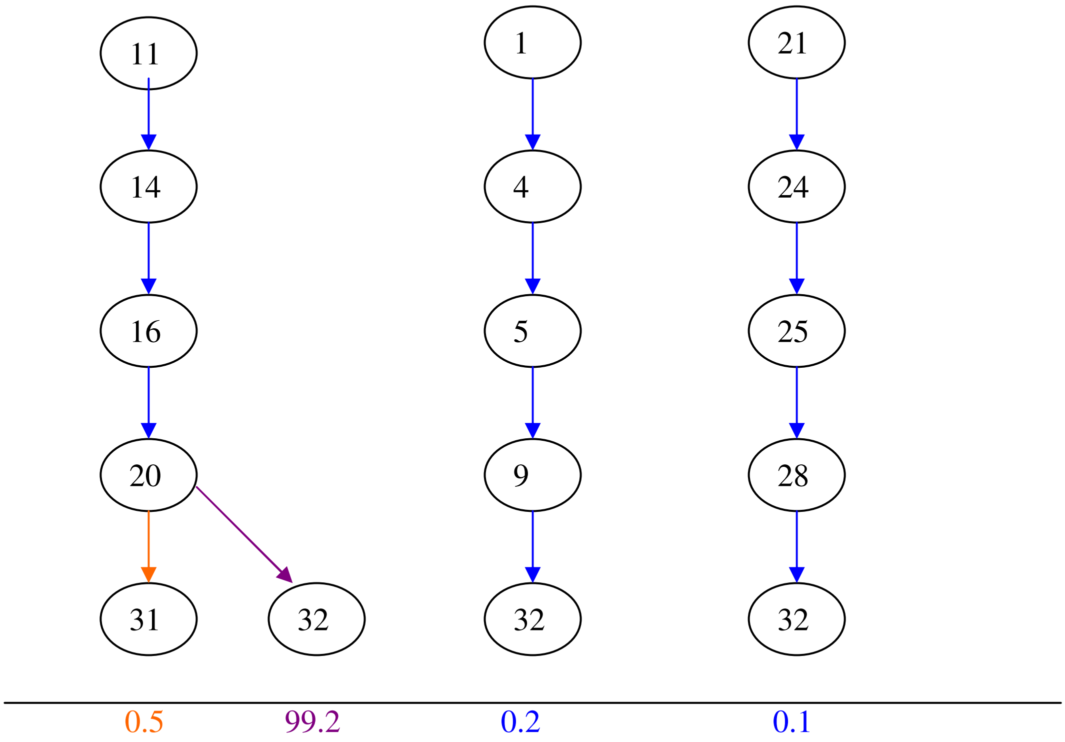


Figure 23: Critical Path Solution Layout for 'Case C'

5.4. Summary of the Cases

The output from all the three cases gives us an indication about how complex the critical path network gets when operational work deviates from actual plans. It also gives us an indication as to how these pre-emptive complex networks can be fruitful for operations planners. For instance in Case A and Case B, activity branches starting from main activity number 1 are too tangled, irrespective of percent occurrence, and represents many different links. Such layout for activities with parallel execution times with a core predecessor that drives them and knowing the level of uncertainty on any one of the branched activities to the predecessor can be helpful from planning perspective.

Shipbuilding encounters several uncertain variables which demands project uncertainty analysis which in turn is useful for shipbuilding/construction scheduling. Project planners use networks with links and nodes with which activities are connected and with which occurrence of activities on the critical path can also be known. This networking with mixed critical paths containing certain probability of occurrence can be helpful in realizing the sequence of changes to work on repetitive operations which CPM is partially vulnerable to.

Knowing the network and how it behaves with distinct uncertain variables is somewhat a strong logic behind avoiding progressing out- of- sequence and avoiding frequent updating of the network if you know which link on the network to be executed if uncertainty grapples the critical one as decided by initial plans.

6. Conclusion

In this thesis, we intended to demonstrate that it is possible to use PERT/CPM together with simulation to suggest alternative planning methodology at Ulstein. From our results, we have shown that simulation with probabilistic distribution gives multiple critical paths depending on the level of uncertainty on the project network. With higher uncertainty and larger project network, it is clear that the number of possible critical paths increases.

The normal PERT/CPM method uses only the activity time means to find the critical path (refer to 3.2.1), thus bringing a deterministic approach to the project and ignores the stochastic nature of the activities. Crashing is the method of reducing project completion time by assigning suitable resources on activities along the critical path of the project network. As we have previously explained (refer to 3.2.2), the probability of achieving reduced project time with conventional PERT network with crashing methods is lower when it has multiple unknown critical paths. Using simulation, we can chart such alternate critical paths and show how probable their occurrences are. This could be used to identify the possible critical paths on which crashing could be performed.

We see that methods such as PERT/CPM and Theory of Constraints assume that there is a constant critical path/bottleneck. In reality, this is not necessarily the case since our results from simulation show that higher degrees of uncertainty on inter-dependent activities actually shift the critical path. Resources at the shipyard could be a bottleneck in one phase of the project, but as well as a slack resource in another.

Based on our experiments with constructed work packages, we see that from planning perspectives, it can be challenging to manage such complex projects as are being conducted at Ulstein Shipyard. We see that Lean Construction (and Lean Shipbuilding) is an attempt to deal with a high degree of variability and uncertainty in a complex reality. We see that by using simulation as a decision-making support tool, we could actually comply with some of the Koskela's Lean Construction principles, hence helping Ulstein to enhance their current Lean efforts. One of these principles is to reduce waste; and it is obvious that waste of man hours is reduced through improved planning: especially in a situation where the precedent activity "A" has to finish before the succeeding activity "B". If the workers assigned to B have to sit idle until A is complete, it is safe to say that this waiting and disruption in flow is wasteful practice according to any lean theory.

According to the shipyard, introduction of the Last Planner system for short-term planning has led to higher degree of co-operation among foremen. This co-operation is said to be important for ensuring correct sequencing of work and resource re-allocation. The 'Pull' planning in Last Planner makes it easier to deal with the complex and uncertain reality because of its flexibility..

Our suggestions of implementing simulation, PERT and CPM would in our opinion strengthen the shipyard's ability to describe uncertainty and re-allocate resources. This is because a planner is required to make estimations on optimistic and pessimistic times in PERT. We also see that simulation has given us information about how frequent activities occur on the critical path. This information enables planners to more easily re-allocate resources from non-critical activities to critical ones (crashing), in an attempt to save time on the project as a whole.

We also assume that humans are not able to take all variables into account in their decision-making, especially in a complex reality; and as a result of this, human-made plans should be sub-optimal compared to a solid computer optimization.

7. Limitations and Further Research

In this thesis, we have explored the interplay between uncertainty, dependencies and duration of activities in short-term planning. The model is restricted from dealing with resources for two reasons. First, it is difficult to describe reality in terms of resources at the shipyard within the scope of this thesis. Second, by dealing with resources it would disrupt our motive of showing a clear correlation between uncertainty and critical paths, in our argument of trying to couple simulation and PERT/CPM.

We have had to rely on assumptions regarding uncertainties and dependencies between activities. In our cases, each of the individual activities has identical degrees of uncertainty. However, it is safe to assume that in reality, different kinds of activities have varying uncertainties because some of the activities are routine work while others are first-timers/customized. This also has implications on how easy it is to estimate the normal duration for different activities. If the shipyard wants to implement a better planning tool they should consider including optimistic and pessimistic durations in their planning as required in PERT.

We suggest development of computerized optimization models which can be used in planning to achieve better performance and this can be experimented by using simulation. If an optimization model is made that incorporated resources and schedules, it can prove that flexible resources such as multi-skilled workers can give shorter duration times than inflexible resources, because the project is subject to uncertainty. This can also have implications on how these workers should be utilized.

8. References

Ballard, Herman Glenn (2000). The Last Planner System of Production Control

Banaszak, Z. A. & Zaremba, M. B. (2006), Project Driven Planning and Scheduling Support for Virtual Manufacturing

Bertelsen, Sven (2000). A Construction Industry Model, at http://cic.vtt.fi/lean/essaybertelsen.htm (accessed 20 May 2008)

Bertelsen, Sven (2008) Lean Shipbuilding – Concept, Theory and Method (6th draft)

Dugnas, Karolis and Uthaug, Ingrid (2007). Can Lean Philosophy Strengthen and Develop Cluster Advantages? – An exploratory Research towards Lean Shipbuilding (Master Thesis at Molde University College)

Dugnas, Karolis (2007). Implementation of LC: A shipbuilding perspective. Presentation at

European Group of Lean Construction, Malmö/Sweden, October 2007

Fishwick. P. A. (1994). Simulation Model Design. Proceedings of the 1994 Winter Simulation Conference: 1-3

Goldatt, Eliyahu and Cox, Jeff (1984). The Goal

Goldratt, Elihahu (1997). Critical Chain

Harrison and Hoek (2005). Logistics Management and Strategy, 2nd edition

Haga, W. A. and Marold. K. A. (2004). A Simulation Approach to the PERT/CPM Time-Cost Trade-Off Problem, Project Management Journal, June 2004: 31-33

Haga, W. A. and O'keefe, T. (2001). Crashing PERT Networks: A Simulation Approach, July 2001: 1-5

Herroelen. W. (2005). Project Scheduling- Theory and Practice, Production and Operations Management, Winter 2005: 413-416

Isaksson, T. (2002). Models for Estimation of Time and Cost based on Risk Evaluation Applied on Tunnel Projects

Kazan. H. (2005). One Application for using PERT Methodology in Strategic Decisions, Journal of Americal Academy of Business, Sept. 2005: 293-296

Kelton, W. David (2003). Simulation with Arena; 2003; Third Edition

Kerzner, H. (2006). Project Management: A Systems Approach to Planning, Scheduling and Controlling : 3-15

Koskela, Lauri (1999). Management of Production in Construction: A Theoretical View

Koskela, Lauri (2000). An Exploration towards a Production Theory and its Application to Construction

Kuklan. H. (1993). Effective Project Management: An Expanded Approach. Journal of Systems Management; March 1993: 1-3

Jeffrey K. Liker and Thomas Lamb (2000): "A guide to lean shipbuilding", at http://www.nsrp.org/projects/deliverables/ase_910001.pdf (accessed: 13 April 2008)

Liker, Jeffrey (2004). "The 14 Principles Of The Toyota Way: An Executive Summary of the Culture Behind TPS", at http://www.si.umich.edu/ICOS/Liker04.pdf (accessed: 13 April 2008)

Liker, Jefferey K. (2004). The Toyota Way

McBride, David (Aug 2003). 'The 7 Manufacturing Wastes', at http://www.emsstrategies.com/dm090203article2.html (accessed 20 May 2008)

Page, W. G. (1989). Using Project management Software in Planning, American Planning Association, Journal of the American Planning Association, Autumn 1989: 494, 495, 498

Pitagorsky, G. (2002). Project management- A Stress Reduction Method: Handling Uncertainty

Poppendieck LLC: "Lean Construction", at http://www.poppendieck.com/construction.htm (accessed 04 December 2007)

Puich. M. (2007). The Critical Path. Biopharm International, Proquest Nursing and Allied Health Source; March 2007: 28-29

Qian, F. & Xu, Y. (2006). Simulation of Schedules in the Less-than-Truckload Trucking Industry

Sunnmørsposten (2008). 'Historical Result in Ulstein", at http://www.smp.no.htest.osl.basefarm.net/article/20080418/NYHETER/804180311 (accessed: 15 May 2008)

The International Group for Lean Construction, at http://www.iglc.net/ (accessed: 5 April 2008)

The Lean Construction Institute, at http://www.leanconstruction.org/ (accessed: 5 April 2008)

The Lean Construction Institute: 4th Annual Lean Project Congress 2002, at http://www.leanconstruction.org/pdf/LCIWebBro9.pdf (accessed 20 May 2008)

Trofin, I. (2004). Impact of Uncertainty on Construction Project performance using Linear Scheduling, University of Florida, 2004: 17-23

Vahkaria. R. P. (2006). Innovative Approaches to CPM Scheduling Management. AACE International Transactions; 2006: 111-114

U.S. Department of Energy. Office of Management, June 2003, at http://management.energy.gov/documents/WorkBreakdownStructure.pdf (accessed: 23 May 2008)

Appendix A

Overview of Entities and Attributes

Entities	**Description**	**Initial Value**
start activity	A dummy entity that checks and generates real entities	
Activity	A real entity that starts on the system and is held by the *Hold* module	
Attributes	**Description**	**Initial Value**
activity_no	Appropriate activity numbers are assigned to the activities released	
Duration	It consist of optimistic, normal and pessimistic durations read from the *ReadWrite* module and held at the *Hold* module	

Overview of Global Variables

Name	**Description**	**Initial Value**
row_counter	It is a logical variable for checking dependencies with value 1 in the rows in the dependency matrix	
num_of_activities	It is control variable for retrieving and keeping track of the value for number of activities. It is set with a minimum initial value of 10 but the real value will be read from the file	10
hold_activity	This variable is a (1D) variable and is used for holding activities that are released through a dummy entity after checking and are ready to enter the system due to dependency less nature	

col_counter	It is a logical variable for checking dependencies with value 1 in the columns in the dependency matrix	
found_a_1	It is used to identify the dependency with binary number 1 and register it, so that the dependency can be traced from this variable and released whenever the system conditions allow	
dummy_counter	The dummy counter is a variable used to keep track of the dummy entities that go through the system and checks for conditions or release of real entities	0
Test	This variable is used to extract and confirm the dependency matrix and duration values to an output file. The purpose is to verify whether the model uses the right data that is fed in	
time used	This variable signifies the time used by an activity to get processed from the system. In other words, time used = time now – start time of the activity	
start time	The variable is a display variable which displays the start time for each activity. This variable is significant for back tracking the critical path for a particular replication	
P_index	It's a logical variable which is automatically generated in Arena from the input file. The dependency table is read from the Excel file and is transformed into a dependency matrix which can be extracted to an output file using the *test* variable and is used throughout on the system	
max_dependencies	It is used to set an upper limit on the	13

	dependencies and it also facilitates in ranging the dependency table to be used on the system from Excel
current_activity	It is a (1D) variable and is used to keep account of current activities being processed on the respective resources
duration_matrix	This variable is used to extract and verify that the duration figures fed in from the input file are correctly used by the system
show_duration	It is (1D) variable and is used as an display variable for animation purpose to show how long the system will take to process the corresponding activity
finish_time_resource	It is also used as a display variable for animation purpose to show how the time tank is emptied while processing the corresponding activity. It also implies as to when the corresponding resource will be available for next use
activity_start_time	It is a (1D) statistical variable through which start time for every activity is extracted to the output file
activity_finish_time	It is a (1D) statistical variable through which finish time for every activity is extracted to the output file. The combination of both start and finish time is important for determining the critical path
critical_path	It is a (1D) variable and through this variable the critical paths for every replication are extracted to the output file
activity_out_counter	Through this variable, the activities that come out of the system are recorded as they come out

finishing_activity	It is a (1D) variable and through it the activities that come out of the system are recorded in the order they finish processing on the system
Dependency	It is a (2D) variable and is used to identify and check the dependencies and release them as their predecessors have finished on the system

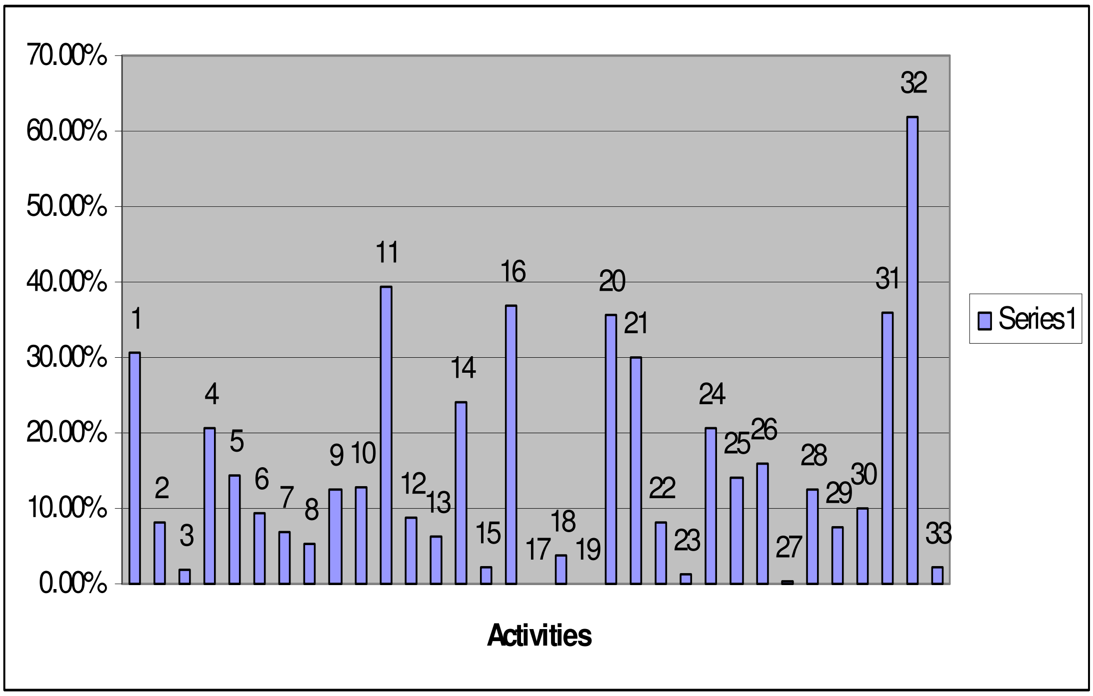
Appendix B

Common Dependency Data for all the Three Cases to be Read by the Model

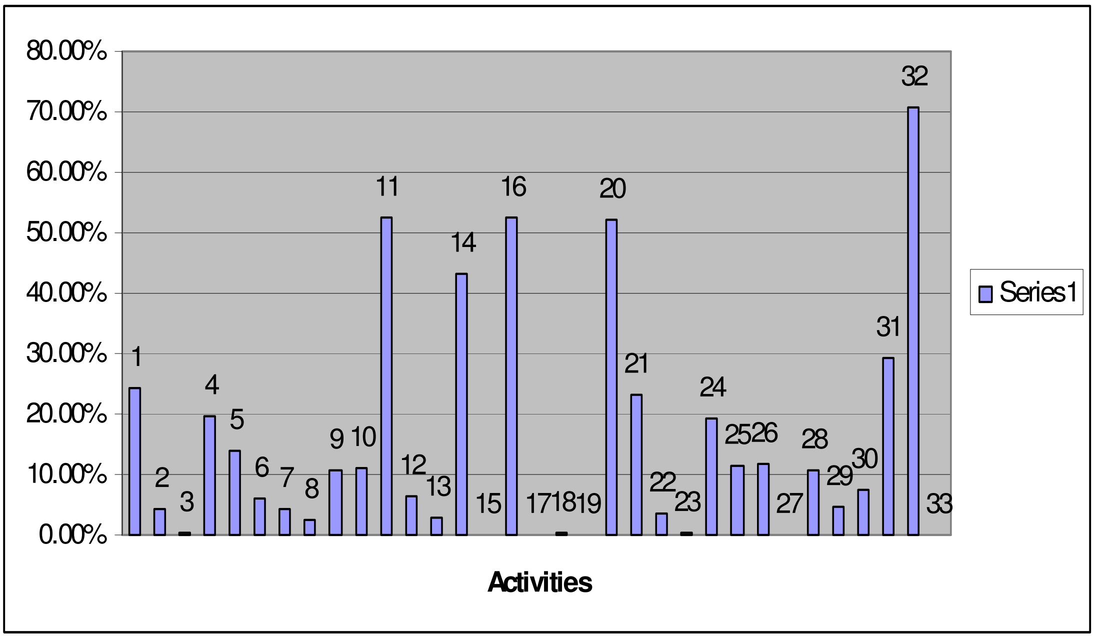
act	d1	d2	d3	D4	D5	d6	D7	d8	d9	d10	d11	d12
1												
2	1											
3	1											
4	1											
5	4	2	3									
6	4	2	3									
7	4	2	3									
8	7	5	6									
9	7	5	6									
10	7	5	6									
11												
12	11											
13	11											
14	11											
15	14	12	13									
16	14	12	13									
17	14	12	13									
18	17	15	16									
19	17	15	16									
20	17	15	16									
21												
22	21											
23	21											
24	21											
25	24	22	23									
26	24	22	23									
27	24	22	23									
28	27	25	26									
29	27	25	26									
30	27	25	26									
31	8	9	10	18	19	20	28	29	30			
32	8	9	10	18	19	20	28	29	30			
33	8	9	10	18	19	20	28	29	30			

Appendix C

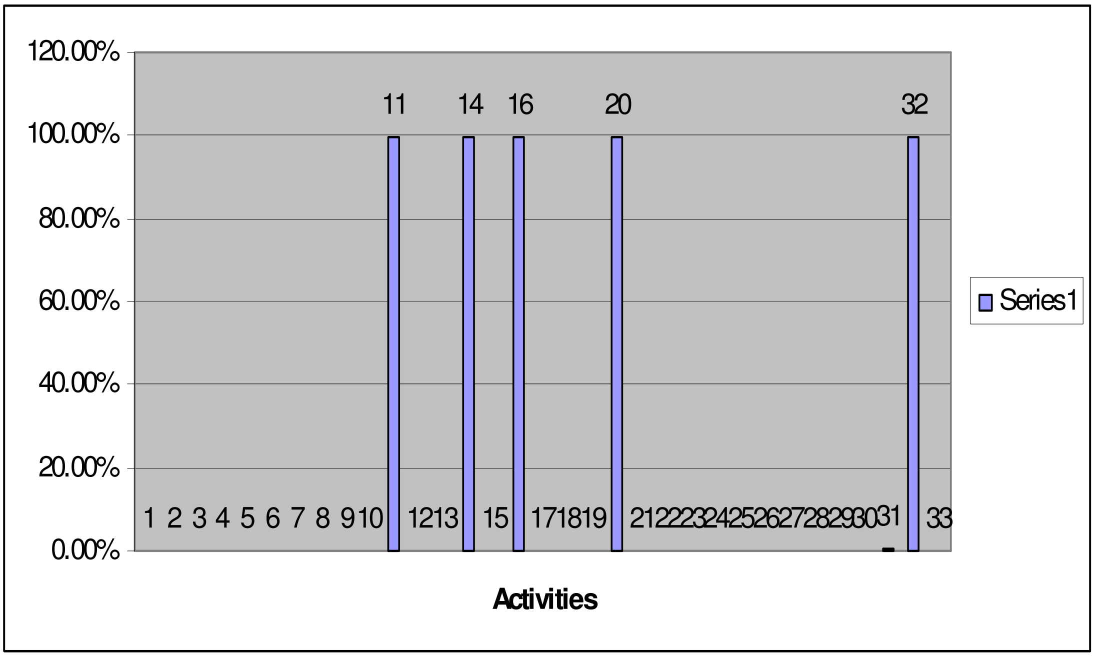
Diagrammatical Representation of the Output obtained regarding Percent Occurrence of Activities on the Critical Path from Case A, B & C



Bar Diagram for Occurrences of Activities on the Critical Path in 'Case A'



Bar Diagram for Occurrences of Activities on the Critical Path in 'Case B'



Bar Diagram for Occurrences of Activities on the Critical Path in 'Case C'

Appendix D

Output for Percent Occurrence of Critical Path from 1000 Replications in Case A, B & C

Important Critical Paths with Critical Path Numbers and Occurrences from 'Case A'

Critical Path Number	Critical Path	Occurrences	% Occurrence
1	11 14 16 20 32	120	12.00%
2	11 14 16 20 31	78	7.80%
3	11 12 16 20 32	42	4.20%
4	11 13 16 20 32	35	3.50%
5	11 12 16 20 31	34	3.40%
6	1 4 5 9 32	30	3.00%
7	21 24 26 28 32	28	2.80%
8	21 24 25 28 32	25	2.50%
9	21 24 25 29 32	24	2.40%
10	21 24 25 30 32	23	2.30%
11	21 24 26 30 32	23	2.30%
12	11 14 16 18 32	19	1.90%
13	1 4 5 10 32	18	1.80%
14	11 13 16 20 31	18	1.80%
15	21 24 26 28 31	18	1.80%
16	1 4 6 9 32	17	1.70%
17	1 4 7 10 32	16	1.60%
18	1 4 5 10 31	14	1.40%
19	1 4 6 10 32	14	1.40%
20	1 2 5 9 32	13	1.30%
21	1 4 5 9 31	13	1.30%
22	1 4 6 10 31	13	1.30%
23	21 24 26 29 31	13	1.30%
24	1 4 5 8 32	12	1.20%
25	21 24 26 29 32	12	1.20%
26	1 2 5 10 32	11	1.10%
27	21 22 26 28 31	11	1.10%
28	21 24 25 28 31	11	1.10%
29	21 24 25 30 31	11	1.10%
30	21 22 26 30 32	10	1.00%
31	1 2 5 9 31	9	0.90%
32	1 4 6 8 32	9	0.90%
33	1 4 7 10 31	9	0.90%
34	21 22 26 28 32	9	0.90%
35	21 24 25 29 31	9	0.90%
36	1 4 7 9 32	8	0.80%
37	11 14 16 18 31	8	0.80%
38	21 22 25 30 32	8	0.80%
39	1 2 6 10 32	7	0.70%
40	11 14 15 20 32	7	0.70%

41	21 22 25 28 31	7	0.70%
42	21 22 25 28 32	7	0.70%
43	21 22 26 29 32	7	0.70%
44	1 2 6 9 32	6	0.60%
45	1 4 7 9 31	6	0.60%
46	11 14 15 20 31	6	0.60%
47	21 22 25 30 31	6	0.60%
48	21 24 26 30 31	6	0.60%
49	1 2 6 8 32	5	0.50%
50	1 4 6 9 31	5	0.50%
51	1 4 7 8 31	5	0.50%
52	11 13 16 18 32	5	0.50%
53	11 14 16 20 33	5	0.50%
54	21 22 25 29 32	5	0.50%
55	1 2 6 10 31	4	0.40%
56	1 2 7 9 32	4	0.40%
57	1 4 5 8 31	4	0.40%
58	1 4 7 8 32	4	0.40%
59	11 12 15 20 31	4	0.40%
60	11 12 16 18 32	4	0.40%
61	21 22 26 29 31	4	0.40%
62	21 22 26 30 31	4	0.40%
63	1 2 5 8 32	3	0.30%
64	1 2 6 8 31	3	0.30%
65	1 3 5 10 31	3	0.30%
66	1 3 5 10 32	3	0.30%
67	1 3 5 9 32	3	0.30%
68	1 4 6 8 31	3	0.30%
69	11 12 15 20 32	3	0.30%
70	21 23 26 28 31	3	0.30%
71	21 23 26 28 32	3	0.30%
72	1 2 5 10 31	2	0.20%
73	1 2 6 9 31	2	0.20%
74	1 2 7 10 31	2	0.20%
75	1 2 7 10 32	2	0.20%
76	1 2 7 10 33	2	0.20%
77	1 3 5 9 31	2	0.20%
78	1 3 6 10 31	2	0.20%
79	1 3 7 10 32	2	0.20%
80	1 4 5 9 33	2	0.20%
81	1 4 7 9 33	2	0.20%
82	11 13 15 20 32	2	0.20%
83	21 23 26 30 31	2	0.20%
84	21 23 26 30 32	2	0.20%
85	21 24 27 30 32	2	0.20%
86	1 2 5 10 33	1	0.10%
87	1 2 7 8 31	1	0.10%
88	1 2 7 8 32	1	0.10%
89	1 2 7 8 33	1	0.10%
90	1 2 7 9 31	1	0.10%
91	1 3 6 10 32	1	0.10%
92	1 3 6 9 32	1	0.10%

93	1 3 7 10 31	1	0.10%
94	1 3 7 9 31	1	0.10%
95	1 4 6 8 33	1	0.10%
96	1 4 7 10 33	1	0.10%
97	11 12 16 20 33	1	0.10%
98	11 12 17 20 32	1	0.10%
99	11 13 16 18 31	1	0.10%
100	11 13 16 20 33	1	0.10%
101	21 22 26 28 33	1	0.10%
102	21 22 26 30 33	1	0.10%
103	21 22 27 30 32	1	0.10%
104	21 23 25 28 31	1	0.10%
105	21 23 25 28 32	1	0.10%
106	21 23 25 30 32	1	0.10%
107	21 24 25 28 33	1	0.10%
108	21 24 26 30 33	1	0.10%
		Sum	**100%**

Important Critical Paths with Critical Path Numbers and Occurrences from 'Case B'

Critical Path Number	Critical Path	Occurrences	% Occurrence
1	11 14 16 20 32	300	30.00%
2	11 14 16 20 31	128	12.80%
3	11 12 16 20 32	44	4.40%
4	1 4 5 9 32	38	3.80%
5	21 24 26 28 32	34	3.40%
6	1 4 5 10 32	33	3.30%
7	21 24 25 28 32	28	2.80%
8	21 24 25 30 32	23	2.30%
9	21 24 26 30 32	23	2.30%
10	11 13 16 20 32	22	2.20%
11	11 12 16 20 31	21	2.10%
12	21 24 25 29 32	18	1.80%
13	1 4 5 9 31	14	1.40%
14	1 4 6 9 32	14	1.40%
15	21 24 25 28 31	14	1.40%
16	21 24 26 28 31	14	1.40%
17	1 4 7 10 32	13	1.30%
18	1 4 5 10 31	12	1.20%
19	1 4 6 10 32	12	1.20%
20	1 4 5 8 32	11	1.10%
21	1 4 6 10 31	11	1.10%
22	1 4 7 9 32	11	1.10%
23	21 24 25 30 31	11	1.10%
24	21 24 26 29 32	11	1.10%
25	1 2 5 9 32	10	1.00%
26	1 2 5 10 32	9	0.90%
27	1 4 7 10 31	8	0.80%
28	11 13 16 20 31	7	0.70%

29	21 22 26 28 32	7	0.70%
30	1 4 6 8 32	6	0.60%
31	21 22 25 28 32	6	0.60%
32	21 24 25 29 31	6	0.60%
33	21 24 26 29 31	6	0.60%
34	21 24 26 30 31	6	0.60%
35	21 22 26 29 32	5	0.50%
36	21 22 26 30 32	5	0.50%
37	1 2 5 9 31	4	0.40%
38	1 4 7 9 31	4	0.40%
39	1 2 6 10 32	3	0.30%
40	1 2 6 9 31	3	0.30%
41	1 4 5 8 31	3	0.30%
42	1 4 6 9 31	3	0.30%
43	11 14 16 18 32	3	0.30%
44	1 2 5 10 31	2	0.20%
45	1 2 6 8 32	2	0.20%
46	1 2 6 9 32	2	0.20%
47	1 2 7 8 32	2	0.20%
48	1 2 7 9 32	2	0.20%
49	1 3 5 10 31	2	0.20%
50	21 22 25 29 32	2	0.20%
51	21 22 25 30 31	2	0.20%
52	21 22 25 30 32	2	0.20%
53	21 22 26 28 31	2	0.20%
54	21 22 26 30 31	2	0.20%
55	1 2 6 10 31	1	0.10%
56	1 2 7 10 31	1	0.10%
57	1 2 7 10 32	1	0.10%
58	1 2 7 9 31	1	0.10%
59	1 3 5 9 31	1	0.10%
60	1 3 6 10 32	1	0.10%
61	1 4 6 8 31	1	0.10%
62	1 4 7 8 32	1	0.10%
63	11 13 16 18 32	1	0.10%
64	11 14 16 18 31	1	0.10%
65	21 22 25 28 31	1	0.10%
66	21 23 25 28 32	1	0.10%
67	21 23 26 28 32	1	0.10%
68	21 23 26 30 32	1	0.10%
		Sum	**100.00%**

Important Critical Paths with Critical Path Numbers and Occurrences from 'Case C'

Critical Path Number	Critical Path	Occurrences	% Occurrence
1	11 14 16 20 32	992	99.20%
2	11 14 16 20 31	5	0.50%
3	1 4 5 9 32	2	0.20%
4	21 24 25 28 32	1	0.10%
		Sum	**100.00%**